ENERGY HARVESTING AWARE ROUTING BASED ON AD-HOC ON-DEMAND DISTANCE VECTOR PROTOCOL

BY

Emad Aldalu

A Thesis Presented to the
DEANSHIP OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

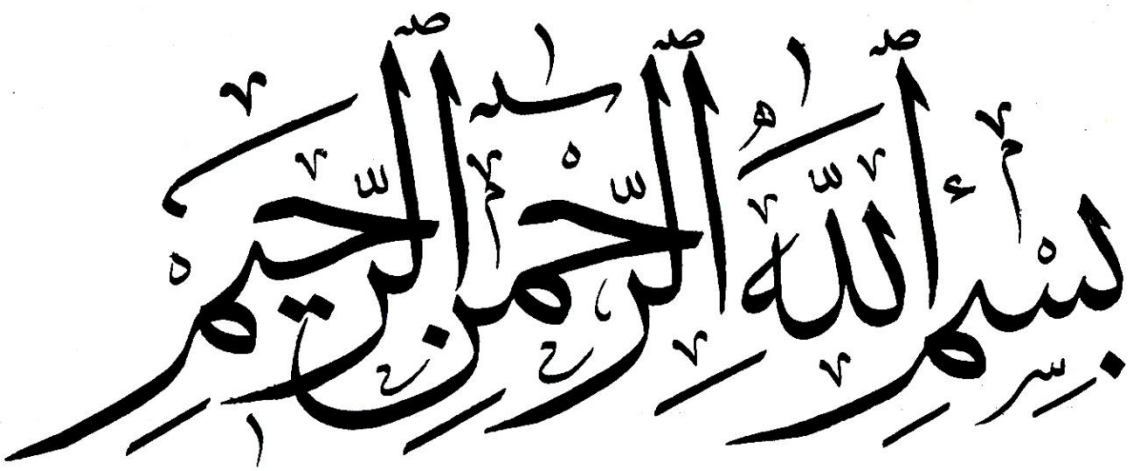
In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

COMPUTER NETWORKS

MAY 2015



IN THE NAME OF ALLAH, THE MOST GRACIOUS AND
THE MOST MERCIFUL

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN- 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis, written by **Emad Aldalu** under the direction his thesis advisor and approved by his thesis committee, has been presented and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE IN COMPUTER NETWORKS.**

Dr. Uthman Baroudi
(Advisor)

Dr. Ahmed Al-Mulhem
Department Chairman

Dr. Shokri Selim
(Member)

Dr. Salam A. Zummo
Dean of Graduate Studies

Dr. Marwan Abu-Amara
(Member)

10/6/15
Date

To my parents, my sisters and brothers

for their endless love, support and encouragement

To my lovely daughter, Dalal

To my lovely son, Mohammad

To my dear wife, Amina

ACKNOWLEDGMENTS

All the praise and thanks to Allah. There is no strength or power except Allah.

I would like to express my gratitude to my advisor, Dr. Uthman Baroudi, for his guidance, expertise, patience, motivation, enthusiasm, and immense knowledge. I would like to thank the members of the thesis committee, Dr. Shokri Selim and Dr. Marwan Abu-Amara for their valuable observations. I would like to thank King Fahd University of Petroleum and Minerals (KFUPM) for granting me this opportunity of pursuing graduate studies.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AODV	:	Ad hoc On-Demand Distance Vector
AOMDV	:	Ad hoc On-Demand Multipath Distance Vector
ACK	:	Acknowledgement packet
EHR	:	Energy Harvesting Rate
DSSC	:	Dye Sensitized Solar Cells
HAN	:	Home Area Network
IP	:	Internet Protocol
IGRP	:	Interior Gateway Routing Protocol
TCP	:	Transmission Control Protocol
IS-IS	:	Intermediate System to Intermediate System
MANET	:	Mobile Ad-hoc Network
MAC	:	Medium Access Control
RREQ	:	Route Request
RERR	:	Route Error
RREP	:	Route Reply
TTL	:	Time to Live

MPPT	:	Maximum Power Point Tracker
MINHP	:	Minimum Harvestable Power
MAXHP	:	Maximum Harvestable Power
Mbps	:	Mega bit per second
OFDM	:	Orthogonal Frequency-Division Multiplexing
OSPF	:	Open Shortest Path First
Kbps	:	Kilo bit per second
QoS	:	Quality of Service
SG	:	Smart Grid
WSN	:	Wireless Sensor Network
WSN-HEAP	:	WSN Powered by Ambient Energy Harvesting

ABSTRACT

Full Name : Emad Ahmad Mahmoud Aldalu

Thesis Title : Energy Harvesting Aware Routing Based On Ad-Hoc On-Demand Distance Vector Protocol

Major Field : Computer Networks

Date of Degree : May 2015

Wireless Sensor Network (WSN) is an evolving technology used in a diverse set of applications. WSN might be applied in isolated or hazard areas where it is hard and costly to reach the nodes. Maintaining the sensor nodes in WSN is generally a complex and difficult task especially when the size of the network is large. Energy harvesting in WSN is a promising technology that enables the sensor nodes to exploit the renewable ambient energy sources such as solar, temperature, sound, pressure, etc. Many data link and network layer protocols were proposed to conserve the energy and use it in efficient way in WSN. In this work, Energy Harvesting Aware Ad-hoc on Demand Distance Vector (EHAODV) routing protocol is proposed. EHAODV is a modified version of AODV. The node's remaining energy, energy harvesting rate and the overall path energy are considered in EHAODV for choosing the best routing path. Moreover, a local route repair mechanism is suggested and implemented in EHAODV. Extensive simulation experiments have been conducted using ns-3 to study the behavior of EHAODV. In comparison with the original AODV, EHAODV has increased the flow goodput about 16%, decreased the packet loss ratio about 13.5%, and increased the packet end-end delay about 8.6%. EHAODV with route repair mechanism has demonstrated lower packet loss ratio around 17% in comparison with AODV.

ملخص الرسالة

الاسم الكامل: عماد أحمد محمود الدلو

عنوان الرسالة: بروتوكول مدرك للطاقة لتوجيه الاتصال في المجسات اللاسلكية بالاعتماد على AODV

التخصص: شبكات حاسوب

تاريخ الدرجة العلمية: شعبان 1436

شبكات الاستشعار اللاسلكية هي تقنية متطورة تستخدم في كثير من التطبيقات. شبكات الاستشعار اللاسلكية من الممكن ان تستخدم في المناطق النائية أو الخطرة بحيث يكون من الصعب الوصول الى المجسات وصيانتها. ان تقنية تجميع الطاقة في الشبكات اللاسلكية لهي تقنية واعدة، تمكن المجسات اللاسلكية من استغلال مصادر الطاقة المتجددة في البيئة المحيطة مثل الطاقة الشمسية و الحرارية و الصوتية و الضغط و غير ذلك و ذلك للحصول على مصدر طاقة متجدد و مستمر. شهدت الساحة العلمية نشاطا كبيرا في تصميم العديد من بروتوكولات الاتصال التي تهدف الى المحافظة على الطاقة واستخدامها بشكل فعال في الشبكات اللاسلكية المعتمدة على الطاقة المتجددة ومع ذلك فهناك مجال كبير للتطوير والتحسين. وفي هذا البحث نقدم بروتوكول اتصال باسم (EHAODV) ويعتمد على البروتوكول المعروف AODV. إن البروتوكول المقترح يأخذ بعين الاعتبار المعايير التالية: كمية الطاقة المتبقية في المجس ومعدل حصاد الطاقة للمجس وكذلك معدل الطاقة المتوفرة في المسار عند اختيار مسار الاتصال الأفضل. إضافة الى ذلك، فإن البروتوكول المقترح يقوم بإصلاح الخلل الآني في مسارات الاتصال. لقد أجريت العديد من تجارب المحاكاة باستخدام NS-3 لدراسة أداء البروتوكول المقترح باستخدام مقاييس أداء متعددة. وبالمقارنة مع AODV، تمكن EHAODV من زيادة سرعة تدفق البيانات بنسبة 16% وتقليل نسبة فقدان البيانات بنسبة 13.5% لكنه زاد الوقت اللازم لوصول البيانات بنسبة 8.6%.

CHAPTER 1

INTRODUCTION

1.1 Wireless Sensor Networks

Wireless sensor network (WSN) is a promising technology emerged before more than a decade [1]. WSNs have various applications in different domains. They are widely used in environmental monitoring applications including but not limited to ground water quality, virtual fencing, and cattle monitoring [2]. In health applications, WSNs are used to monitor biological and physical signals of the human body [3]. WSNs are also used in military applications such as perimeter protection, urban warfare and overnight village monitoring [4]. Figure 1.1 shows a typical example of WSN.

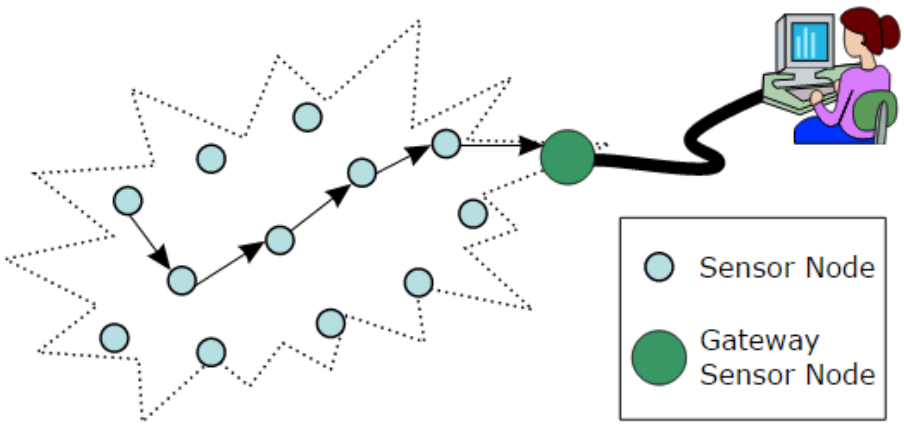


Figure 1.1 Typical multi-hop wireless sensor network architecture [5].

The nodes in WSN relay the monitored data between each other until it reach the sink node. The sink node aggregates the received data and then sends it to a remote station for further analysis.

1.2 Routing Protocols

Routing is one of the essential components of a computer network's functionality. In a broad sense, it is the process of determining a path to a desired destination. For example, a network administrator could compute all the routes in the network and then deploy them to all nodes manually. However, this approach is impractical and very error-prone for even a small network. As a result, a large number of routing protocols have been created that dynamically determine the routes in the network without human involvement. Without routing protocols, routing information would have to be manually and continually maintained by network administrators; networks would have been unable to scale to the size that it must in order to operate within modern day networks.

After route discovery, the data can be delivered to various destinations in the network on a hop-by-hop basis. As a packet travels through the network, each intermediate node (called a "hop") independently determines the interface on which the packet has to be sent out in order to reach its destination. This process is known as forwarding. It utilizes a data structure known as a forwarding table, which maps a network or node address to the next hop on the route to a particular destination. Forwarding is achieved by consulting the forwarding table to determine the next node (i.e., the outgoing interface) to which a data packet must be sent in order to eventually arrive at the desired destination. This process is repeated at each intermediate node on the packet's path.

The primary intent of any routing protocol is to build and maintain the routing and forwarding tables. A routing protocol is a set of rules that defines how nodes should interoperate and exchange data in order to build routing tables and ultimately achieve efficient end-to-end forwarding. Some popular routing protocols are Open Shortest Path First (OSPF), Intermediate System to Intermediate System (IS-IS), and Interior Gateway Routing Protocol (IGRP).

1.2.1 Link-State and Distance Vector Protocols

Generally, routing protocols are classified into one of two categories: Link-State or Distance-Vector [6]. In link-state routing algorithms, each node acts independently to map the entire network topology as a graph. This is achieved by sending information about each node's neighbors to all other nodes in the network. Each node then uses the graph of the network topology it has created to compute the most efficient route to each destination in the network.

In contrast to link-state algorithms, distance-vector algorithms do not distribute routing information to all the nodes in the network. Instead, each node only informs its neighbors about topological changes in the network. These changes are represented as an array of distance-vectors (i.e., known distances from the current node to each node in the network). As a result, distance-vector algorithms are considered computationally simpler and create less control traffic overhead than do the link-state algorithms.

1.2.2 Ad-Hoc Networks

An ad-hoc network is a network that can operate in an environment without preexisting infrastructure and allow for minimal configuration during deployment [7]. In

an ad-hoc network, each node acts independently both as an end node and as a router. Each node may send, receive, and forward data traffic. These attributes are very desirable in areas where a network is needed but in which no prior infrastructure exists (e.g., search and rescue, disaster relief systems, military operations).

A Mobile Ad-hoc Network (MANET) is a wireless ad-hoc network in which the nodes can move and change their location [8]. Consider an area that was just stricken by a natural disaster (e.g., hurricane, tornado, tsunami). Natural disasters can wipe out existing networking infrastructure, making communication in the affected area difficult or impossible. The ability of a MANET to function in areas with no prior infrastructure makes them extremely useful in disaster scenarios. However, mobility and lack of infrastructure also present an interesting problem in terms of routing, for which specialized routing protocols are needed.

1.2.3 Proactive and Reactive Routing Strategies

Routing protocols in ad-hoc networks can be classified into three different categories: proactive, reactive, or hybrid [9]. Proactive routing protocols will actively seek routes to destination nodes, even if there is no traffic traveling through the network. A proactive routing protocol seeks routes in anticipation that they will be needed later. The advantage of proactive routing protocols is that routes are readily available as soon as there is data to be transmitted. However, they may result in unnecessary overhead when searching for routes that will never be used. In a MANET environment where computing resource and bandwidth are scarce, this could be a major deficiency that will prevent the protocols from being widely deployed.

On the other hand, reactive protocols only compute routes on demand (i.e., only when a node has data to transmit and the path to the destination is unknown). The advantage of reactive routing protocols is that routes are only computed when they are needed, which minimizes the amount of control overhead introduced into the network. However, the data has to wait while the routing protocol searches for the route. Hybrid routing protocols achieve optimal performance by combining the advantages of reactive and proactive approaches.

1.3 Ad-hoc On-Demand Distance Vector

Ad-hoc On-Demand Distance Vector (AODV), as the name implies, is a reactive distance-vector routing protocol. AODV consists of two primary phases: route discovery and route maintenance. The route maintenance phase is responsible for removing outdated or broken path entries from the routing table.

1.3.1 Route Discovery

AODV only initiates route discovery when a node, often referred to as the originator, receives data from the application layer that is to be delivered to some destination for which there is no known route. The originator starts the route discovery phase by broadcasting a Route Request (RREQ) message. The RREQ message is rebroadcast by each intermediate node until it reaches either the destination node or a node with a fresh route to the destination. At that point, the node generates a Route Reply (RREP) message back to the originator. The route discovery phase terminates when an RREP message that contains a route to the destination arrives at the originator node. As the RREP traverses the network back to the originator node, it retraces the path of the RREQ message, which was recorded by the intermediate nodes as the RREQ message

was traveling through the network. Similarly, intermediate nodes that receive an RREP message update their routing tables with the route to the destination node. In MANETs, if some intermediate nodes have moved away before forwarding the RREP, this behavior causes the route discovery to fail and a new route discovery process to be initiated. Once the route discovery phase completes, the originator node sends data to its destination over the newly discovered path.

Once a route has been discovered and stored, it will only remain in a routing table for a finite amount of time. When a route is stored, it is initially marked as active. Active routes will remain useable either for the Lifetime value received in the RREP message or for a minimum preconfigured default time period. When the timer eventually expires, the routes are marked for deletion and are scheduled to be removed from the routing table.

1.3.2 AODV Control Messages

AODV uses three types of control packets during the route discovery phase. These control packets are Route Request (RREQ), Route Reply (RREP), and Route Error (RERR) [10]. An RREQ packet is used anytime AODV needs to discover a route to a specific node. An RREP is used to reply to an RREQ with a definitive route to the node. RERR packets are used to disseminate various error details to other nodes in the network. AODV maintains route entries of its active one-hop neighbors by periodically broadcasting Hello messages with the IP header TTL field set to one. Hello messages have the same format as the RREP messages and can carry the IP address and the destination sequence number for the current node. The sequence number is a unique counter created and maintained by each node. This value is included in all messages that carry routing information. The sequence number represents the freshness of carried data

and also prevents routing loops. An AODV node with multiple routes to the same destination is required to select the freshest route (i.e., the route that has the largest destination sequence value). An AODV node increments its sequence number each time it initiates a new route discovery process and whenever it generates an RREP message. This ensures that other nodes in the network can differentiate between RREP messages generated from different route request phases.

1.3.3 Expanding Ring Search

In order to reduce the overall control message overhead, AODV employs an expanding ring search technique. The originator node sets the TTL field in the IP header of the RREQ message to a certain initial value. If the route discovery process fails to find a path to the destination, then the originator node increments the value of the TTL field and repeats the process again. This continues until either the originator node finds a path to the destination or the whole network has been searched without finding a path (i.e., an RREQ message with IP TTL field set to the preconfigured TTL threshold value was sent out, but a route to the destination was not found). The originator node maintains a timer (NET_TRAVERSAL_TIME) in which the RREP message is expected to be received within. If the timer is expired before receiving the RREP message, the originator mat try again to discover a route by broadcasting another RREQ, up to a maximum of RREQ_RETRIES times at the maximum TTL value. This search technique prevents unnecessary network-wide dissemination of RREQs.

Despite the expanding ring search technique, the route discovery process in AODV often results in a large number of control packets traveling through the network. This consumes already scarce network resources (e.g., bandwidth, processing power, battery

power). Furthermore, anytime there is a demand for a route that is either marked for deletion or does not exist in the routing table, the routing protocol must rediscover the path. This can be costly in volatile MANETs, as the constant change in the topology of the network frequently causes routes to become unavailable. Continuously re-computing routes creates substantial overhead in the network, which will eventually lead to performance degradation.

1.4 Problem Statement and Contribution

Sensor nodes are normally deployed in high numbers in situations where it is hard to supply them with power lines. Traditionally, sensor nodes are powered with on-board batteries that limit the operational lifetime of the nodes. Batteries need to be changed or recharged periodically which is expensive and a time consuming process. Alternative energy sources for WSN are recently investigated. In this work, we specifically address the energy harvesting technique. Energy harvesting techniques gather the energy from the ambient energy sources that are present in the environment like solar, thermal, vibration energy, etc. and convert that energy into a form that can be used to power the devices. Using energy harvesting techniques in WSNs has many advantages over batteries. First, the node can operate perpetually without need for human intervention. Second, energy harvesting techniques are friendly to the environment whilst depending on high capacity batteries pollute the environment [11].

WSNs are widely used in Smart Grids (SGs), also called intelligent power grid. WSNs provide a feasible and cost-effective sensing and communication platform for remote system monitoring and diagnostic. Using sensor network in SGs has to satisfy special requirements; the information generated by sensor networks may be associated

with some data QoS requirements, such as reliability, latency, and network throughput. For example, the critical sensed data related to grid failures should be received by the controller in a timely manner [12]. The communication subsystem supporting sensor networks must provide mechanisms to satisfy these QoS requirements.

Routing in WSN is one of the techniques that are used to conserve the energy [13-16]. Many energy conservation protocols were proposed and tested in the literature. The energy harvesting Home Area Network (HAN) that is deployed in SGs has special characteristics. The nodes in this network have no mobility and the energy source is renewable. A node in energy harvesting network stops its operation when its energy is discharged and comes back to work when it harvests enough energy. This behavior has similar effect on the network topology as mobility does. The difference is that if the nodes never die then the network is static i.e. no mobility effect. If these network characteristics are well exploited by a customized routing protocol, the network can meet QoSs that the general routing protocols fail to do. The work in this thesis aims to develop a routing protocol that takes the node's remaining energy, energy harvesting capability, and route path sum of energy in consideration in building the routes. In addition, a route repair technique is also implemented. Following is the list of contributions of this thesis:

- Modify the control plane of AODV protocol to collect the nodes energy information including the minimum energy in the route, the minimum energy harvesting rate in the route, and the sum of energy of all nodes in the path.
- Develop a route selection criteria based on the energy information where the minimum energy has the top priority.

- Develop a route selection criteria based on the energy information where the sum of energy has the top priority.
- Develop a local route repair mechanism by discovering a bypass route.
- Compare the performance of EHAODV protocol with original AODV protocol by testing them in ns-3 simulator.

Chapter 2 provides an overview of energy harvesting techniques and energy harvesting model of ns-3 simulation package. Chapter 3 discusses the related work of energy-aware routing protocols. Chapter 4 describes the comparison study of fixed battery and energy harvesting sources in WSN using AODV protocol. Chapter 5 discusses in detail our proposed energy harvesting aware protocol called EHAODV. Chapter 6 describes the comparison study of EHAODV and the original AODV protocol using ns-3. The thesis is concluded in Chapter 7.

CHAPTER 2

ENERGY HARVESTING

In this chapter we present the common energy harvesting techniques in WSNs and also present the energy harvesting model of ns-3 simulation package.

2.1 Energy Harvesting Techniques

Wireless sensor networks powered by ambient energy harvesting (WSN-HEAP) is a concept in which the sensor node uses one or more energy harvesting module to harvest the ambient energy such as light, vibration and heat from the environment and store the harvested energy in a capacitor [11]. The differences in the system architecture between a battery-powered wireless sensor node and WSN-HEAP node are shown in Figure 2.1

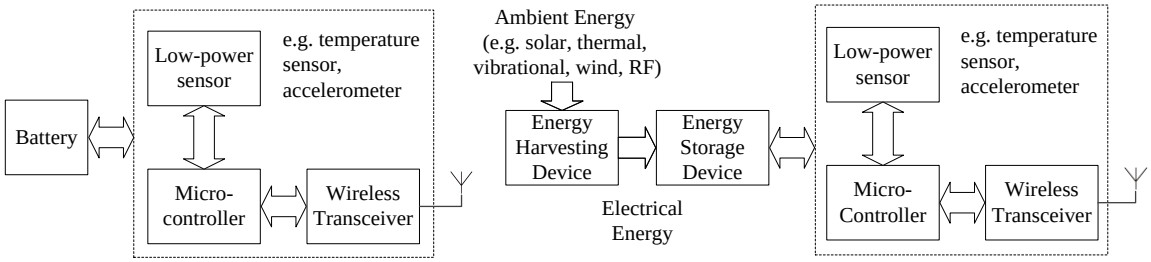


Figure 2.1 Battery-operated wireless sensor versus WSN-HEAP node [17].

The goal of WSN-HEAP is to maximize the throughput and minimize the delay since the energy is renewable and the concept of network lifetime does not exist. On the other hand, the throughput and delay are usually traded off in battery-powered networks to achieve a longer network lifetime. The protocol design in WSN-HEAP is designed such

that the node is alternating between sleep and wakeup modes in order to save the harvested energy as much as possible. The harvesting rate of a node varies across time and the type of the energy harvester.

The renewable energy technology is not new; while using it with small devices as sensors is a new challenge. The common renewable energy techniques include but not limited to hydro electric power generation using water, wind turbines, solar panels, and thermal sources. The renewable energy technology has to be comparable in size with the sensors. The most appropriate energy harvesting technologies for WSNs are solar, mechanical, and thermal.

2.1.1 Solar Energy Harvesting Technique

Among the different energy harvesting techniques, solar power is the most spreading and mature one. The downside of solar power is that it is capable of generating energy only with the existence of sun or artificial light. On the other hand, the existing solar power systems are traditionally not designed to be used with small scale and low energy consumption systems such as WSNs. Recently, advancements in the photovoltaic technology are introduced such that solar cells with area close to one cm^2 are available [18]. A photovoltaic system called GaAs, which is a high efficiency solar cell designed for indoor light. Its power density was found to be over 3x greater than Dye sensitized solar cells (DSSC) modules under indoor light levels [19]. An energy prediction model named Pro-Energy was proposed in [20]. The model leverages the past energy observations of multiple harvesting sources and provide accurate estimation of future energy availability. The model was tested on real life solar and wind traces. A solar energy harvesting circuit using an adaptive maximum power point tracker (MPPT) that

ensures that the energy is not lost during the transfer from the harvester to the energy storage device for 1-mW solar-powered mote was presented in [21]. The proposed circuit achieved fast transient response, small steady state oscillation, and low power consumption.

2.1.2 Mechanical Energy Harvesting Technique

The movement and vibration and other forms of kinetic energy of objects can be exploited to generate electric energy. Vibrations exist in bridges and roads because of the vehicles movement. It also exists at the body operating motors and engines [22]. Piezoelectric energy conversion is commonly used method to transform the mechanical energy into electrical energy [23]. An energy harvesting shoe system for podiatric sensing was introduced in [24]. The system generates from 10 to 20 μJ per step.

2.1.3 Thermal Energy Harvesting Technique

The temperature difference between two ends of a conductor results in a heat flow which results in diffusion of charge carriers. The flow of charge carriers between the hot and cold regions in turn creates a voltage difference. An energy harvesting circuit uses hybrid of indoor light and thermal was proposed in [25] to extend the lifetime of sensor nodes. The circuit harvested an average of 620 μW at an average indoor solar irradiance of 1010 lux and a thermal gradient of 10 degrees. The area of the solar panel used is about 16 cm^2. In [26], an thermoelectric energy harvester was integrated into a shirt and tested on people in real life. It generated power in the range of 0.5 to 5 mW at ambient temperatures of 15 °C to 27 °C, respectively.

The authors of [11] identified that the total harvested energy of the state of the art harvesters is not enough to power the sensor node continuously. Table 2.1 shows different energy harvesting technologies and its corresponding harvesting rates with 10 cm^2 harvesting material. The achievable duty cycle of Crossbow MICAz sensor according to each harvester is shown.

Table 2.1 Achievable duty cycle by MICAz with 10 squared cm harvesting material.

Technology	Energy Harvesting Rate (mW)	Duty Cycle %
Vibration - electromagnetic	0.04	0.05
Vibration - piezoelectric	5	6
Vibration - electrostatic	0.038	0.05
Thermoelectric	0.6	0.72
Solar - direct sunlight	37	45
Solar - indoor	0.032	0.04

2.2 Energy Harvesting Model

In order to simulate a network that considers the nodes energy consumption, an energy model is needed. Network Simulator 3 (ns-3) has an energy model described in [27] which was added to ns-3 version 3.9. The energy framework of ns-3 consists of two basic models. The first one is the energy source model which represents the node energy supply source. Examples of this is linear and Li-ion battery models. The other one is the device energy model, which represents the energy consumption of the node such as Wi-Fi radio model. An energy-harvesting model was recently implemented in ns-3 version 3.21. This model allows the energy source remaining energy to increase based on predefined configurations. A uniformly distributed random amount of energy is added to the remaining energy of the energy source every second. There are two attributes that affect

the harvesting rate; minimum harvestable power (minhp) and maximum harvestable power (maxhp). The actual harvested energy every second is a uniformly distributed value that lies between minhp and maxhp:

$$\mathrm{EH(t)} = \text{ Uniform } [\mathrm{minhp}, \mathrm{maxhp}] \tag{1}$$

The available energy in a node in idle time i.e. no energy consumption, can be represented as a function of time as following:

$$AE(t) = \sum_{0}^{t} EH_t + IE\ ; \qquad t = 0, 1, 2\ \ldots \tag{2}$$

Where AE(t) is the available energy at time t, IE is the initial energy. In order to verify the energy-harvesting model, we have conducted a simulation experiment where we traced the available energy in a node. minhp and maxhp values were set to 0 and 4.57 mW respectively. Figure 2.2 shows two curves representing the available energy of the node.

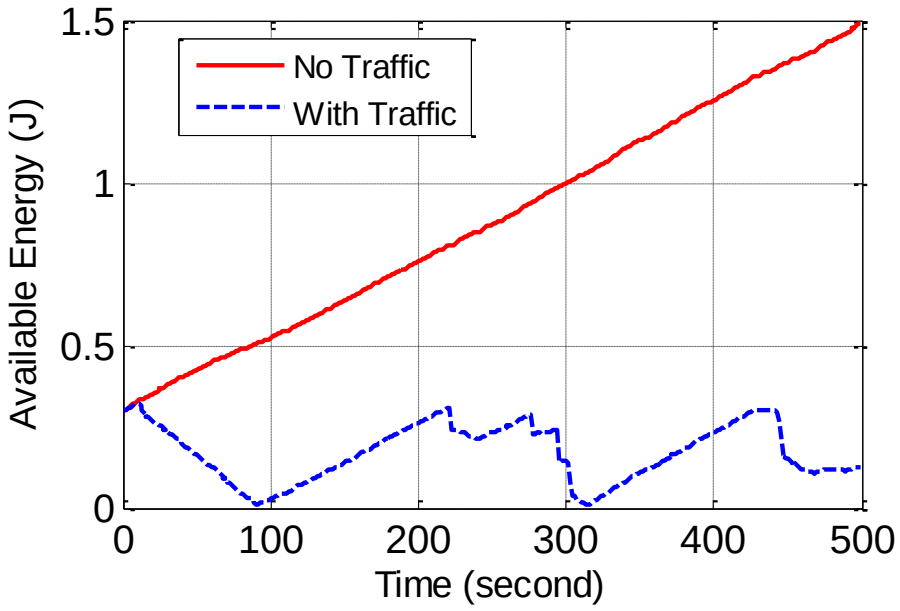


Figure 2.2 Available energy of the node.

The upper curve represents the available energy of the node where no traffic is carried in the network. Therefore the node is not consuming energy. The mean value of the harvesting rate of the node equals 2.29 mW. If the average harvesting rate is multiplied by the total time of the simulation which is 500 seconds; the result equals 1.14 J which is the total harvested energy by that node. The initial energy is 0.3 J, so the total available energy at the end of the simulation is close to 1.5 J. The lower curve shows the available energy of the node in an active network. The node harvests energy and at the same time consumes energy for sending and receiving packets. The node started with 0.3 J and started communicating. Close to time 100 s, the node has depleted all its energy. When the node energy reaches zero, the node is turned off for a period of time, enough to allow the node to harvest amount of energy equals RiseTh. RiseTh is set to 0.3, this means that the node remains off until it harvests a sum of 0.3 J. Close to time 210 s, the node energy exceeded 0.3 J so it was turned on. After that, the node started communicating and its

energy was decreasing. Close to time 310 s the node was turned off again. The process of turning the node on and off based on its energy level continues until the end of the simulation. It is clear that the node consumes energy more than it harvests.

CHAPTER 3

LITERATURE REVIEW

In this chapter we survey the related work in energy aware and energy-harvesting aware routing protocols in WSNs.

Doost et al. [28] proposed a routing metric based on the charging ability of the sensor nodes. The proposed metric aims to prolong the network lifetime by forming the routes with nodes that have the best energy charging characteristics. The method includes an optimization function that derives 1) the node charging time (T_{ch}) in which the node don't transmit data but only charges its capacitor, 2) node transmission time (T_{tx}) in which the node is allowed to send and receive data. A new routing metric based on the node charging time is proposed. Charging time is defined as the time needed for the node battery to be fully charged. The node also measures the T_{ch} standard deviation over multiple trials. AODV protocol was modified based on this approach as following. RREQ packet was modified to include the tuple $< T_{ch}Max(k), N_{ch}Max(k) >$ that travels over path k. $T_{ch}Max$ represents the maximum charging time considering all the nodes in path k. $N_{ch}Max$ is the associated standard deviation of the maximum charging time value. A node might update $T_{ch}Max$ value if its own T_{ch} is greater than $T_{ch}Max$. Each node in the path introduces a forwarding delay of the RREQ based on its own charging rate. The delay equals to the sum of the charging time and its standard deviation divided by a factor. This ensures that the RREQ of best routes arrives at the destination first. At the destination, multiple RREQs are received from multiple paths. The destination waits for

certain period of time then chooses the path with the lowest value of charging time and sends RREP message through that chosen path. This work supports single path routing where the best path is chosen by the destination. Moreover, the energy harvesting technique in this work is limited to wireless electromagnetic waves.

Tan and Bose [29] proposed a power aware routing protocol for MANETs based on AODV called PAW-AODV. The routes in PAW-AODV are chosen based on a power-based cost function. The cost function of a route is the sum of the cost functions of the individual nodes along the route, where the cost function of individual node depends on the available battery power of the node. The aim of PAW-AODV is to choose a route r with least cost. The proposed cost function is defined as in Eq. (1) and Eq. (2). $C_i(t)$ is the cost function of node i at time t. $b_i(t)$ is the available battery power of node i at time t.

$$C_i(t) = \frac{1}{b_i(t)} \quad \text{for node } n_i \tag{1}$$

$$c(r,t) = \sum c_i(t) \text{ for all nodes } n_i \text{ that lie on route } r \tag{2}$$

Tcr_{RREQ} is a new field added to RREQ control message to store the route cost as the RREQ packet propagates in the network. Tcr_{RREP} is a new field added to RREP control message to store the route cost from the source to destination. Similarly, Acr_{RREP} field is added to RREP message to compute the forward route cost at each node. The node cost is divided into four zones based on the available battery power where each zone is assigned a fixed cost. The node cost doesn't change instantaneously as the available power changes. Instead, the cost changes if the available power moves from one zone to another. This reduces the overhead of the route cost maintenance. PAW-AODV is a

single path routing protocol. It considers the energy in general but not the energy harvesting.

Lotfi et al. [30] proposed a routing protocol based on power-aware cost function. Additional variables were added to the route request message to collect necessary information throughout the network and make decision about routing. Request size ($reqSize$) field represents the size of data that the source node is going to send. *Unstable Nodes Count* field holds the number of unstable nodes which has a rate of change in the number of neighbors higher than a threshold. *Sum of Neighbors* field holds the sum of neighbors of all nodes a cross the path. *Sum of Buffered Packets* field holds the sum of buffered packets in all nodes a cross the path. Each node calculates its remaining life time by dividing it residual energy by the average energy consumption per second. Nodes with remaining life time less than the time needed to send the data packets drops the route request packet and hence leads to constructing a new path. The cost is calculated at the destination as a sum of weighted values of the above three fields. The destination sends a route reply message throughout the path of least cost. The routing protocol in this work is a single path. The best path is chosen at the destination. It doesn't utilize the collected energy information for route maintenance.

Poongkuzhali et al. [31] proposed Optimized Power Reactive Routing (OPRR) protocol for MANETs based on AODV. The technique includes adding a new field to the RREQ message. The new field maintains the available power of all neighboring nodes of the entire path. While the RREQ is propagating, each node adds the sum of available power of its neighboring nodes to the power field. The intermediate node that receives the RREQ and has a forward path to the destination is not allowed to respond by a RREP.

Instead, the RREP is always initiated by the destination node. The destination might receive the RREQ from multiple paths. The destination calculates the average power of each path based on the power field and the number of hops. The destination replies with a RREP message for the path with highest average power and lowest hop count. The problem of this approach is that the average path power alone is not sufficient to identify the best path. According to this approach, a best path could consist of a node that has very low energy. In addition, this approach doesn't utilize the routing information in the intermediate nodes and it is a single path routing protocol.

Alshanyour and Baroudi [32] proposed Bypass-AODV, which is a local recovery protocol to enhance the performance of AODV. Instead of reconstructing a full route on the occurrence of route errors, Bypass-AODV uses cross-layer MAC-notification to identify mobility-related link break, and then setup a bypass between the broken-link end nodes via an alternative node while keeps on the rest of the route. The simulation results showed that Bypass-AODV outperformed original AODV in terms of throughput and packet drop ratio. If used in energy harvesting network, Bypass-AODV can meet QoS such as network availability and low latency.

The authors of [33] proposed an Enhanced Ad-Hoc On-demand Distance Vector routing protocol (EAODV). EAODV searches for paths that consist of nodes that have the most proper transmitting power. Also it chooses the minimal power consumption route through comparing the existent routes. The results showed that EAODV reduced the transmitting power and also reduced the collisions.

Authors of [34] proposed Energy Harvesting Aware Ad hoc On-Demand Distance Vector Routing Protocol (AODVEHA) that make use of the energy harvesting capability of the sensor nodes in the network, The protocol reduced the packet delivery energy cost in comparison to original AODV.

Table 3.1 shows the different routing metrics of the different protocols that we surveyed in the literature. In EHAODV, three metrics were used in addition to route repair mechanism. The remaining energy of the node and the average energy in the path are directly obtained from the node, it don't need for history information to be collected. While the energy harvesting rate differs from time to another, so a moving average of history readings can be used. In our simulation we used a uniform distribution for the energy harvesting rate so we used the mean value. The remaining node life time metric can't be obtained directly from the node. This metric depends on other parameters such that the node remaining energy, node harvesting rate, and the traffic load on the node. Request size metric help in making a decision whether to send data now or wait based on the energy information in the path.

Table 3.1 Different routing metrics which used in the energy-aware protocols in the literature.

Protocol	**Metric**					
	Remaining energy	Energy harvesting rate	Average energy	Route repair	Remaining lifetime	Request size
EHAODV	✓	✓	✓	✓	×	×
Doost et al.	✓	✓	×	×	×	×
Tan, Bose	✓	×	×	×	×	×
Lotfi et al.	✓	×	×	×	✓	✓
Poongkuzhali et al.	✓	×	✓	×	×	×
Gong et al.	×	✓	×	×	×	×

We didn't find in the literature a routing protocol for WSN that has all the characteristics that are targeted in this work. The target of this work is to come up with specially designed routing protocol for energy harvesting based networks that targets QoS such as high throughput, low packet loss ratio, and low network latency. The proposed protocol exploits the nodes remaining energy, energy harvesting rate, and path sum of energy in building the routes. In addition, the proposed protocol consists of a route repair technique.

CHAPTER 4

AODV UNDER ENERGY HARVESTING

AODV is a routing protocol that was adopted by Zigbee for mesh network [35]. However, not much work exists in literature that explores deeply the capabilities of AODV under energy harvesting. In this chapter, extensive simulation experiments have been conducted using ns-3 simulator to study the behavior of AODV for energy-harvested and limited-battery WSN assuming total supply energy is the same for both approaches. The purpose of this chapter is to test the behavior of AODV under energy harvesting environment and to understand the effect of turning the sensor nodes on and off on the performance of the network. We have compared AODV for both approaches using different performance metrics.

4.1 Simulation Setup

Network simulator 3 (ns-3) version 3.21 was used to simulate the network. ns-3 is a discrete-event simulator used in research and education [36]. It is open source software written in C++. The simulation parameters of the test environment are summarized in Table 4.1. We have considered two simulation scenarios: energy harvesting and fixed battery.

Table 4.1 Simulation Parameters.

Parameter	Value
Test Area	560m X 560m
Number of nodes	81
Placement	9 X 9 grid
Radio range	100 m
Nodes separation	70 m
Transmission bandwidth	6Mbps
Traffic type	Continuous rate of 2 Kbps, TCP
Number of sources	Varying 2-12
Packet payload size	300 Bytes
Sink	Single sink node at the center
Source to destination physical distance	4 hops
Routing protocol	AODV
Simulation time	2000 sec
Scenario replications	40

In both scenarios, there are 81 nodes placed in 9X9 square grid pattern as shown in Figure 4.1. The nodes are geographically fixed i.e. no mobility is considered in the network as the nodes power outages have a similar effect on the network as the mobility does. A node reaches eight other nodes directly if it is at the center and three other nodes if it is at the corner. The network topology, number of nodes, and radio range parameters are close to related work in [29]. The energy model parameters are discussed in section 6.1.1. The sink is placed at the center of the network so it has a highest number of neighbors and to assure fair energy consumption in all the regions in the network.

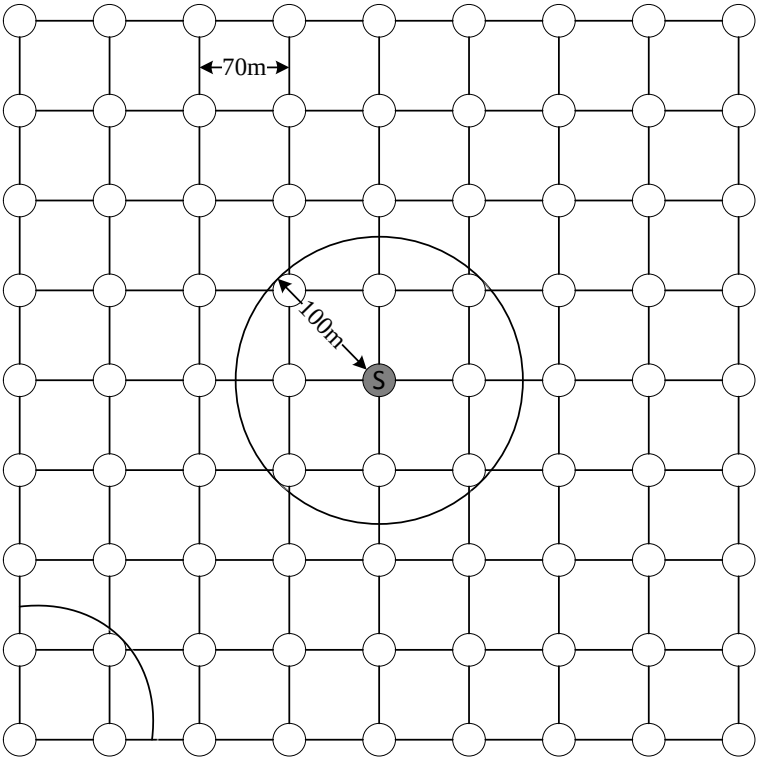


Figure 4.1 Nodes placement.

The Sink node is fixed at the center to guarantee fair energy consumption in all of the nodes. The source nodes are randomly chosen from the edges. Hence the hop-count of the shortest path from any source to the destination is four. AODV protocol was used in both scenarios as layer-3 routing protocol. The source node starts sending TCP traffic to the sink as soon as there is a valid route to it. We assume that each source has exhaustive packets, so data transmission will only stop when the simulation ends. In addition, both scenarios are assumed to leverage the same total energy budget. For example, if we have n nodes with n batteries and each battery has 1J, then we have nJ as a starting budget for the battery based scenario. For the energy-harvesting scenario, the simulation continues until the total of nJ is harvested. This way makes our comparison fair where each scenario has the same energy budget.

4.1.1 Energy-harvesting Scenario

In this scenario the nodes depend on the harvested energy in order to operate. The nodes are supplied with 0.3 J as initial energy. This amount of energy is enough for the nodes to initiate the communication. The energy harvesting rate of the nodes is uniformly distributed between 0 to 4.5 mW. That is each node is expected to harvest 2.25 mW. For a simulation time of 1000 seconds, the expected harvested energy of the node is 2.25 J. The total available energy for each node equals the harvested energy plus the initial energy. Therefore, the available energy of each node is 2.55 J. Whenever the energy of a node reaches zero, it is turned off so that it stops sending or receiving any data, but it continues harvesting energy. If a node is turned off, then the source node will construct a new route when needed. When a dead node (i.e. turned off) harvests a sum of 0.3 ($RiseTh$) J, it is turned on, so that it can participate in the communication. $RiseTh$ stands for the setting rise threshold. The initial energy, energy harvesting rate, and *RiseTh* are chosen such that the nodes available energy is not always low and not always high. Low energy available energy causes a high number of power outages in the network and high available energy causes few of power outages.

4.1.2 Fixed Energy Scenario

In this scenario the nodes depend on a fixed battery as energy source. The nodes don't harvest any energy in this scenario. The nodes are supplied with 2.6 J at the beginning of the simulation. After starting the simulation, the nodes start communicating and hence losing its energy until they die. When a node dies, it can't participate in the communication for remaining simulation. If a dead node was participating in an active route, the nodes in that route reconstruct a new route to the sink. Both scenarios were run

with 2, 4, 6, 8, 10 and 12 sources concurrently. Increasing the number of sources will increase the total traffic in the network. Each simulation was replicated 40 times with randomly selected source nodes in order to reach 90% confidence interval. The replications are independent as each simulation has different seed value.

4.2 Performance Metrics

The energy consumption is proportional to the number of sources. More sources in the network means more traffic is carried in the network. In this work, a power outage is defined as a node goes from on state to off state. With low number of sources, the nodes power outages are rare. Ns-3 software package has a flow monitor module that provides a flexible system to measure the performance of the network. The module uses probes, installed in network nodes, to track the packets exchanged by the nodes, and it will measure a number of parameters. Packets are divided according to the flow they belong to, where each flow is defined according to the probe's characteristics. For IP, a flow is defined as the packets with the same protocol, source IP, source port, destination IP, and destination port tuple. The probes measure the packet bytes including IP headers while layer-2 headers are not included [37]. In order to evaluate both scenarios and other scenarios in this thesis; we shall use all or some of the following performance metrics:

1. Total Received Bytes at the Sink

This metric traces the number of received bytes (transport layer) at the sink node from all source nodes. The total received bytes increases as the number of sources increases.

2. Packet Loss Ratio

For each TCP flow, it traces the number of transmitted packets at the source and the number of received packets at the sink. Packet lost ratio equals the number of lost packets divided by the number of sent packets. The packet loss ratio of the simulation equals the average packet loss ratio for all TCP flows.

3. End-to-End Delay

It is the time that a packet takes to travel from the source until it reaches the sink node. This metric is calculated for all the transmitted packets then the average value is traced.

4. Total Consumed Energy

It is the total energy consumed by all of nodes during the simulation. The total consumed energy is proportional to the total sent/received packets.

5. Total Number of Alive Nodes around the Sink

This metric traces the state of the neighboring nodes of the sink whether they are on or off.

6. Average Goodput per Flow

Throughput is defined as the quantity of error-free data that is transmitted per unit of time. Goodput is defined as the application layer throughput [38]. This metric traces the number of received bytes per flow and divides it over the total flow time.

7. Total Number of Power Outages

This metric traces the total number of power outages in the network across the entire simulation.

4.3 Simulation Results

As we mentioned above, we will discuss the simulation results for five performance metrics for each scenario. The simulation was replicated for 40 times at each point. The confidence intervals were calculated at 90% confidence level.

Figure 4.2 shows the total received bytes at the sink. When the number of sources is less than four, the network is able to operate with rare outages. The sink node received almost the same number of bytes in both scenarios. When the number of sources increased to twelve, more energy was consumed which caused more power outages in energy-harvesting scenario and higher number of nodes die in fixed-energy scenario. The sink node in the energy-harvesting scenario received twice the number of bytes as the sink received in the fixed energy scenario.

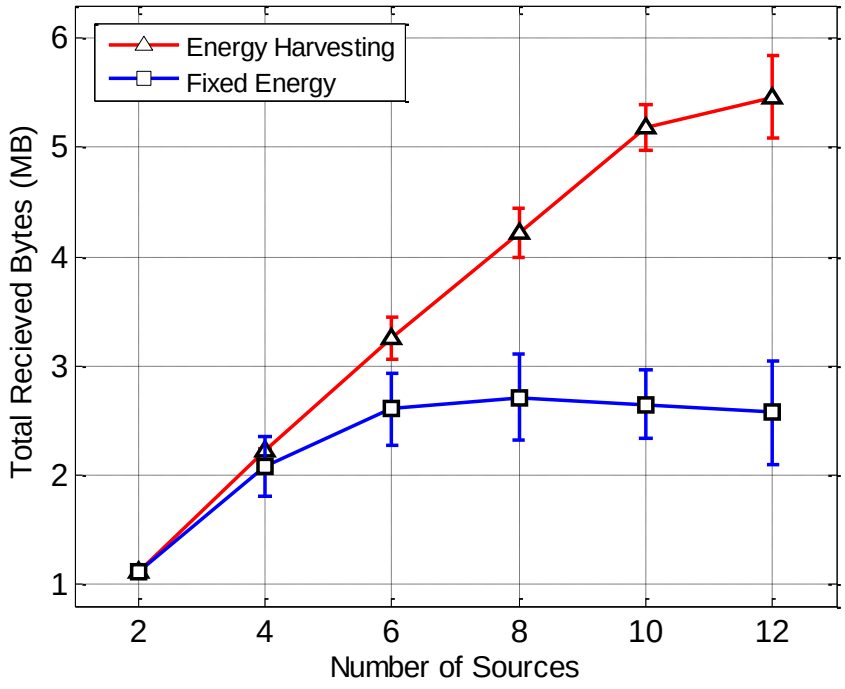


Figure 4.2 Total number of received bytes at the sink.

The amount of energy in the fixed-energy scenario is limited. When the number of sources is low, they successfully sent their data to the sink. When the number of sources is higher than six, the network is not capable to serve all the requests because of the power outages, so the total received bytes at the sink was not increasing.

Figure 4.3 shows the packet loss ratio. As the number of power outages is rare at low number of sources, the packet loss ratio was close to each other in both scenarios. The larger the number of sources, the more power outages are experienced by the routes. These power outages cause the packets in the intermediate nodes to be lost. The packet loss ratio for fixed energy scenario is doubled when twelve TCP connections are running simultaneously.

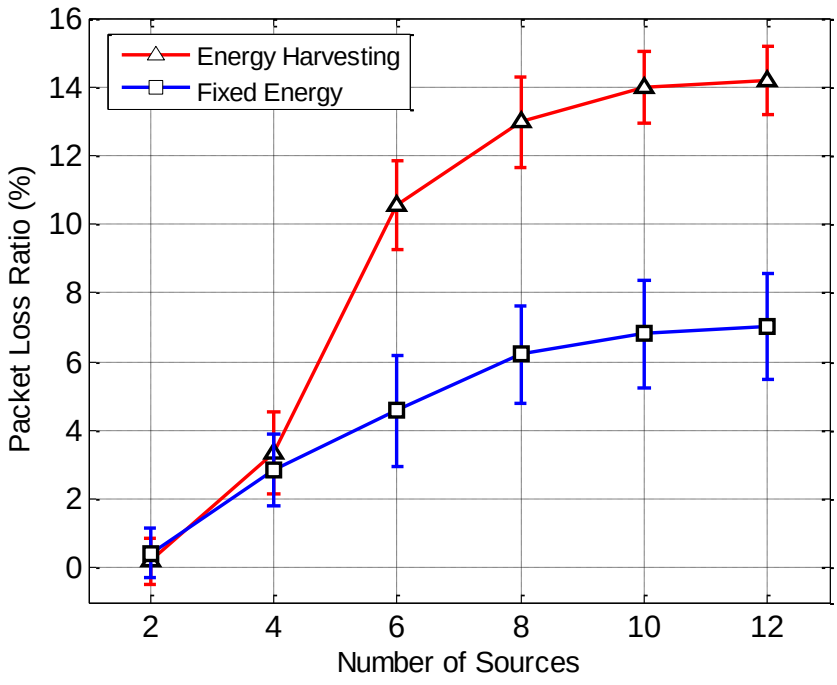


Figure 4.3 Average packet loss ratio per flow.

On the other hand, AODV with energy-harvesting scenario has experienced higher delay than fixed-energy scenario as shown in Figure 4.4. Both scenarios behave the same when small number of TCP sessions are running. For instance, when there are twelve TCP connections, the average end-to-end delay for energy-harvesting scenario is about 29 ms with about 23% increment compared to fixed energy scenario.

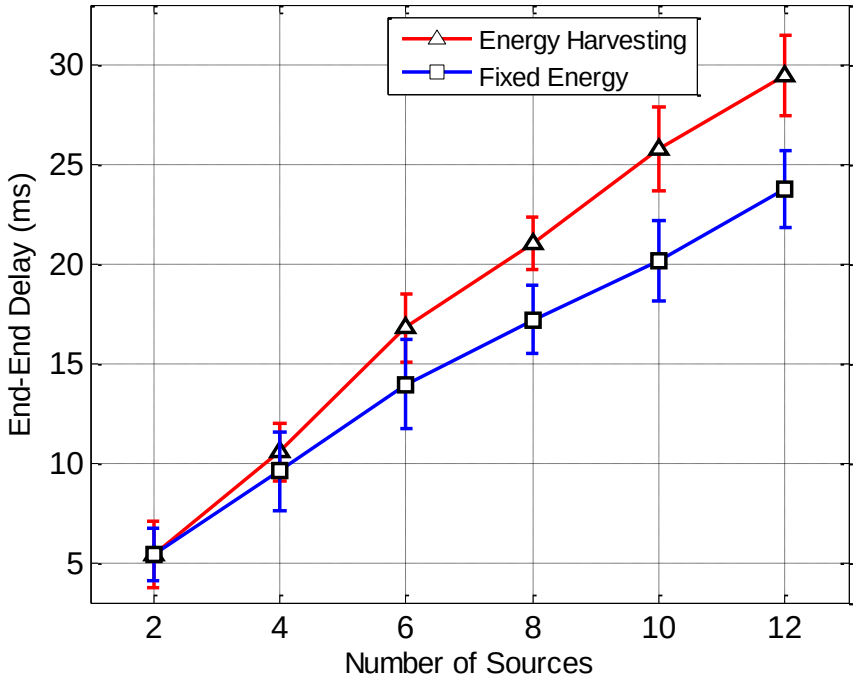


Figure 4.4 Average packet end-to-end delay per flow.

Figure 4.5 shows the total consumed energy in Joules by all nodes. As the number of sources increases, more packets were transferred in the network, which means more energy is consumed. The consumed energy at two sources is close to each other as the total received bytes are similar at the same points. Another factor affects the consumed energy; it is the route hop-count from the source to destination. Higher hop-count route consumes more energy than lower hop-count routes. As a result, the total received bytes

might be equal but the consumed energy could be different. When the number of sources increases, power outages become imminent under energy-harvesting scenario. This results in packet loss and packet retransmission. Even the total received bytes at number of sources of six for energy harvesting is twice the fixed energy, but the consumed energy is the same for both scenarios at the same point. At higher number of sources, the consumed energy for energy harvesting scenario is roughly 30% higher than the consumed energy of fixed energy. This is because the total received bytes is higher for energy harvesting scenario.

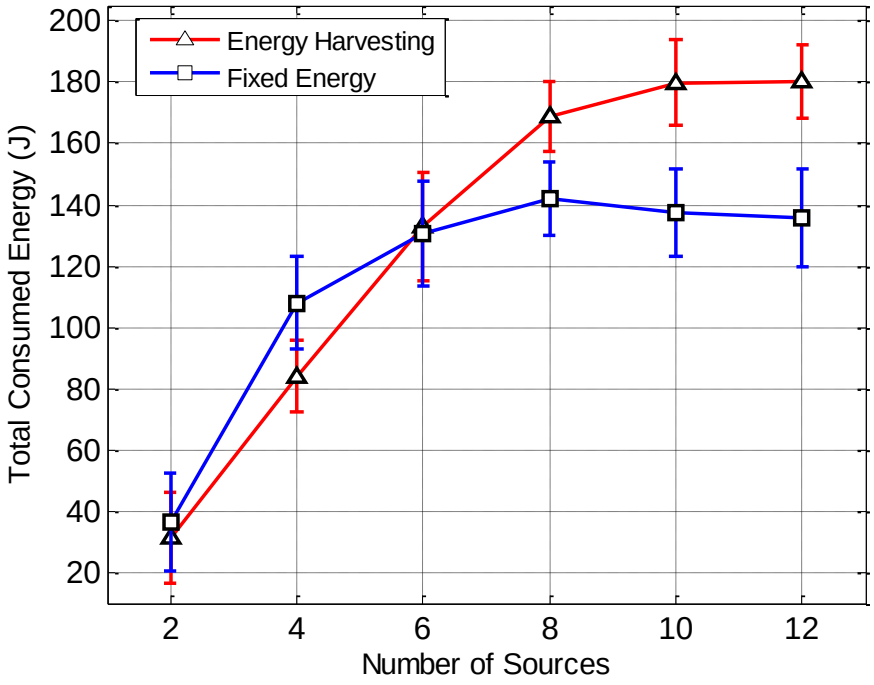


Figure 4.5 Total consumed energy of all nodes.

Finally, the nodes in the energy-harvesting scenario consumed the energy in efficient way. Suppose that a neighboring node is listening to the traffic but it is not part of the route, turning off this node will not affect the data session but will result in lower total

consumed energy. This situation explains why the energy-harvesting scenario is more efficient than the fixed energy scenario in using the energy. In order to understand why the sink received more packets in the energy harvesting-scenario than the fixed energy scenario, the number of alive nodes that surrounds the sink versus time is shown in Figure 4.6. As shown in Figure 4.1, the sink node is located at the center of the network. There are a maximum of eight active neighboring nodes. At the beginning of the simulation, all of the nodes are working in both scenarios as seen in Figure 4.6.

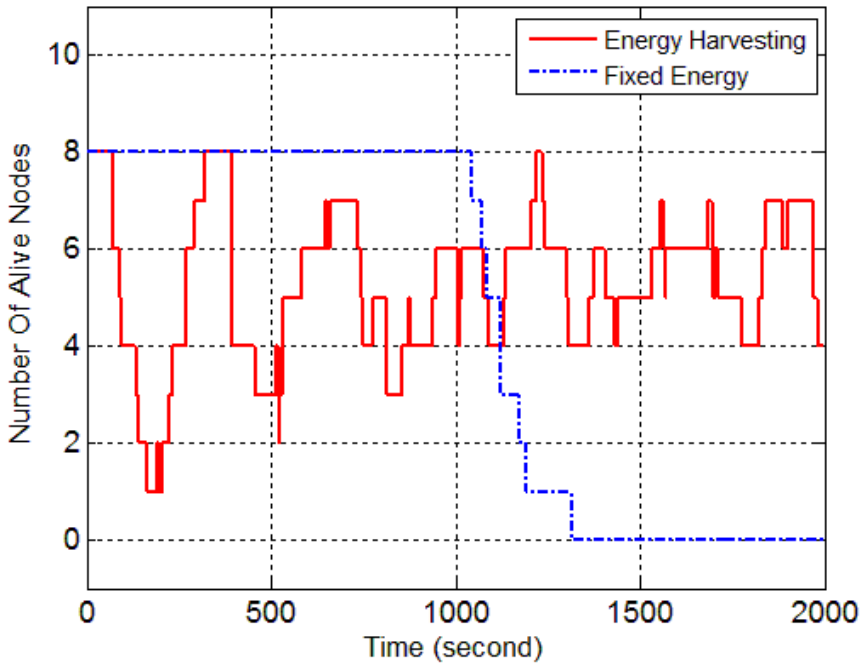


Figure 4.6 Number of alive nodes around the sink.

The neighboring nodes in the energy-harvesting scenario start dying before completing the first 100s. Around time 200s, seven neighboring nodes were turned off. In the fixed energy scenario, it takes much longer time until the sink's neighboring nodes start going down. Around time instant 1300s, all the neighboring nodes were down. For the fixed energy scenario, when all the sink's neighboring nodes are down, the source nodes

cannot construct a path to the sink so that no more data can reach the sink. On the other hand, the sink in the energy-harvesting scenario is never disconnected. While in the fixed energy scenario it was disconnected in the last 700s of the simulation. It is worth to note that the sink's surrounding nodes exhibit a cyclic behavior where it rarely happens that all nodes are down. This interesting finding is due to the energy statistical multiplexing where not all surrounding nodes have sufficient energy at the same time. Consequently, these nodes are alternating this status (active/turned-off) such that there is always at least one active node that connects the sink to the rest of the network.

4.4 Conclusion

Energy harvesting based WSN is getting a lot of momentum and attention because of its ability to provide continuous supply of power. Nevertheless, the path for widely proliferation in industrial applications still needs to overcome several challenges such as supporting QoS routing. The energy harvesting scenario outperformed the fixed energy scenario in the total received bytes at the sink. While the fixed energy scenario outperformed the energy harvesting scenario in the packet loss ratio and end-end delay. This result can be explained if we understand how the simulation goes. In the fixed energy scenario, the sources send their packets without severe problems until the nodes energy start going down. A power outage in the fixed scenario can't be repaired later. The neighboring nodes of the sink transmit the higher number of packets so they are the first nodes which are turned off. When all neighboring nodes are turned off, the sink node which is at the center is disconnected from the network so it can't receive any more packets. In the energy harvesting scenario, the power outages starts early as the nodes have low initial energy. But the power outages in the energy harvesting scenario are

repairable. If a node can't construct a route to the sink, after some time it will be able to do that as some nodes are turned on. Energy harvesting aware route protocols that chooses the best route based on the energy and the harvesting rate of the nodes can decrease the total outages in the energy harvesting scenario.

CHAPTER 5

ENERGY HARVESTING AWARE AD-HOC ON-DEMAND DISTANCE VECTOR

The routing protocol is responsible for building the communication paths from the source to destination. Ad hoc On demand Distance Vector (AODV) is a well-known routing protocol widely used in WSN and smart grid (SG) communication [39]. AODV supports both unicast and broadcast routing [40]. Our solution is based on modifying AODV protocol to collect the node remaining energy and energy harvesting information along the route discovery path and using this information in addition to the hop-count in the route selection criteria. The goals of collecting the energy information are as follows: 1) exclude nodes, which have energy level less than a pre specified threshold from being a part of any routing path, 2) choose nodes, which have sufficient energy level to carry the communication session.

5.1 Modifying AODV to Collect Node's Energy Information

The node's energy level and energy harvesting rate play a major role in the proposed protocol; Energy Harvesting Aware Ad-Hoc On-Demand Distance Vector (EHAODV). The node has a battery where the node saves its energy in it. Zero battery level indicates that the node energy is fully discharged. The battery capacity is assumed to be high enough so that the node can save all its harvested energy. The node battery level varies continuously based on the node activity. Each time the node receives or sends

a packet, it loses some energy. Depending on the availability of ambient energy, the node charges its battery. It is essential to optimize the routing so that the harvested energy doesn't get wasted. We propose adding three fields to RREQ and RREP packets of AODV. The first field is *minEnergy*, which stores the minimum available energy of the nodes along the path in which the RREQ or the RREP is propagating. The second field is *sumEnergy*, which stores the sum of the available energy of all nodes along the path in which the RREQ or the RREP is propagating. The third field is *minHarRate*, which stores the minimum energy harvesting rate of the nodes along the path in which the RREQ or the RREP is propagating. These three fields along with the hop count will be used as criteria to choose the best route. Figure 5.1 shows the proposed format of RREQ. The added fields are shown in bold.

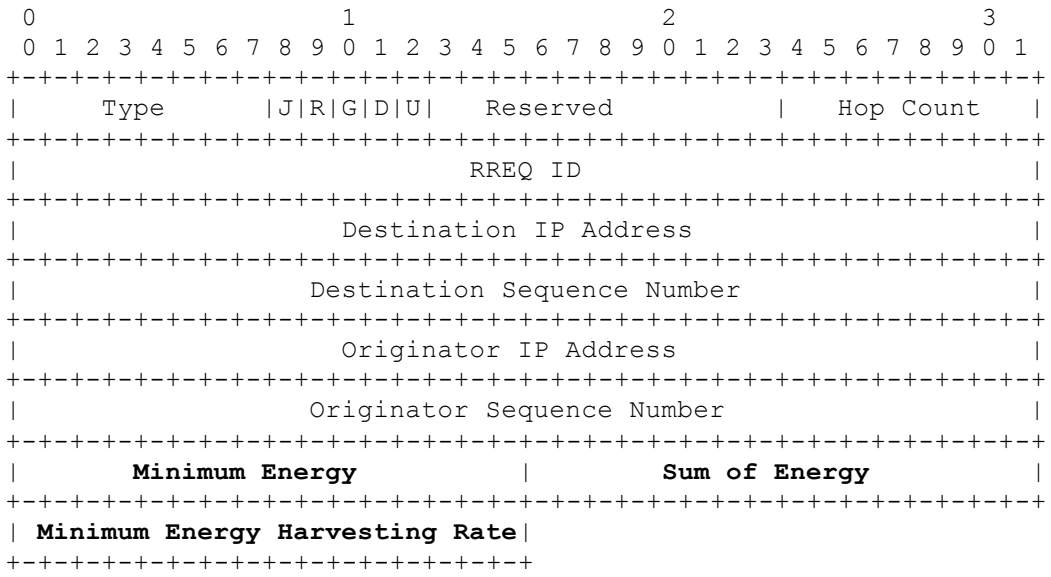


Figure 5.1 RREQ format of EHAODV [10].

The proposed fields are added to RREQ message in order to trace the energy information in the reverse route entries while the RREQ is propagating and added to the RREP

message to trace the energy information in the forward routing entries while the RREP is traveling to the originator node.

Figure 5.2 shows the proposed format of RREP. The added fields are shown in bold.

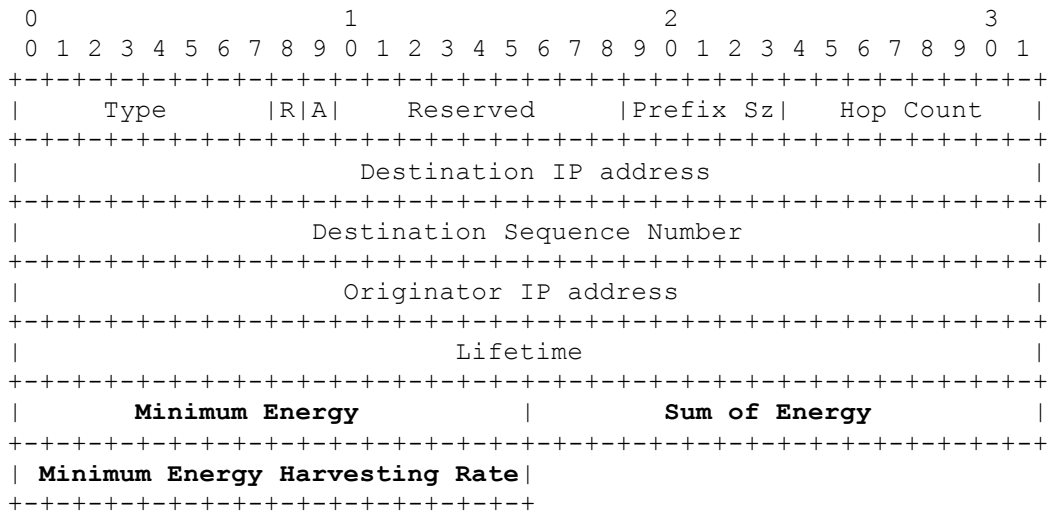


Figure 5.2 RREP format of EHAODV [10].

When the source node has data packet destined to a remote node, it checks its routing table whether a route already exists for that node or no. If no route exists for the destination, the source begins a route discovery process. The route discovery process is initiated by the source node by broadcasting a RREQ message which is then flooded in the network. The destination IP address in the RREQ is set to the IP address of the destination node. Figure 5.3 explains how the node deals with the energy fields. The route entries have a timeout value that indicating the freshness of the route. Initially, *minEnergy* and *sumEnergy* are set to the source remaining energy and *minHarRate* is set to the source energy harvesting rate. Once the RREQ reaches a neighboring node, the node builds a reverse route to the originator. Then it compares its remaining energy with *minEnergy* filed in the RREQ message. If the node's remaining energy is less than

minEnergy field, it updates the RREQ with its energy. Otherwise, it leaves the value as it is. The node adds its remaining energy to *sumEnergy* field in the RREQ and update the field with the sum. The process of updating *minHarRate* is similar to *minEnergy*. The node updates the field if its energy harvesting rate is less than the RREQ field, else it keep the value as it is. Then the node checks if it is the desired destination. If not, it broadcasts the RREQ to its neighboring node. This process continues until the RREQ reaches the desired destination.

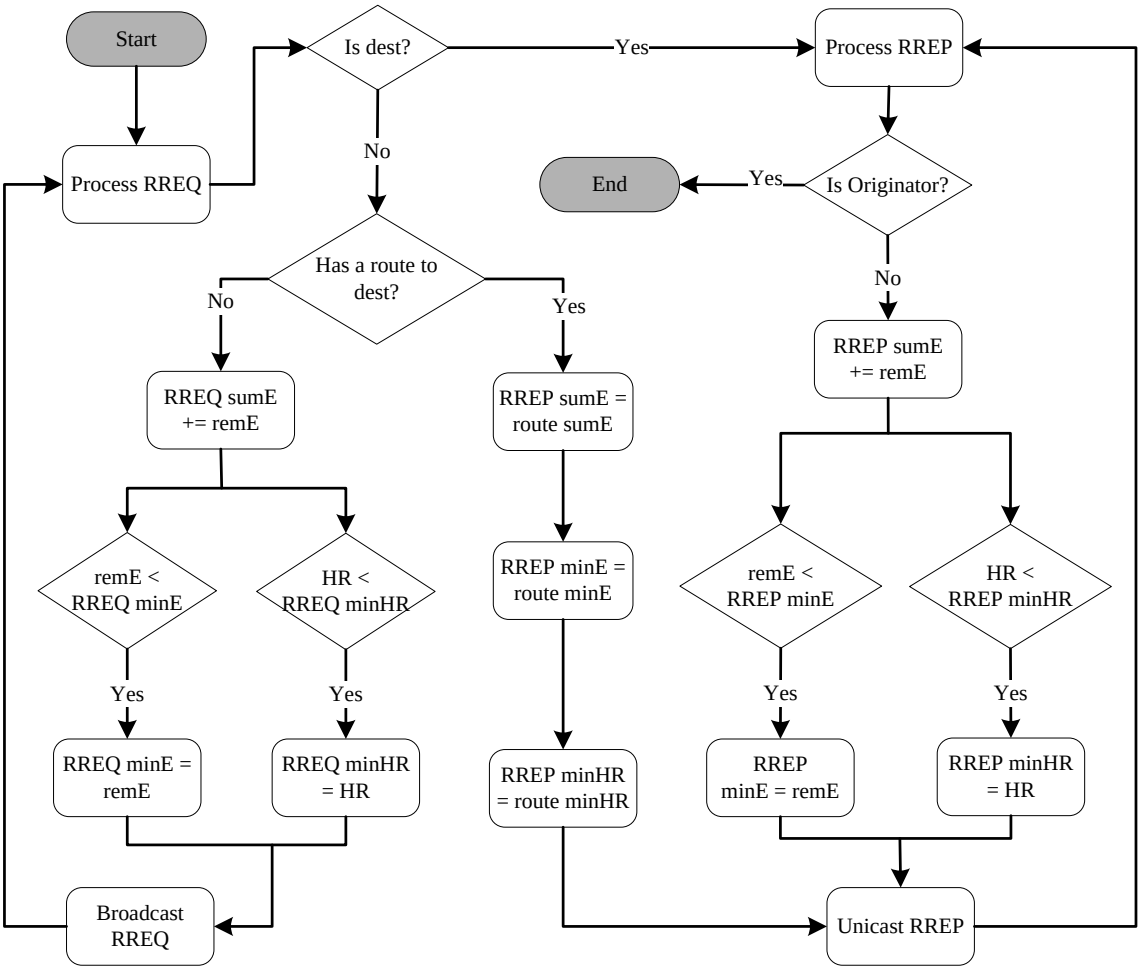


Figure 5.3 Energy fields update process in EHAODV.

When an intermediate node receives a RREQ to a destination that it already has a valid route to it, the intermediate node responds with a RREP message. The node compares its remaining energy with the minimum energy in the route entry and sets the minimum of them in the RREP message. The node does the same thing for the minimum harvesting rate field. It also adds its remaining energy to the sum of energy of the existing route and sets the sum in the RREP field. When the destination node receives a RREQ it replies with a RREP message where minimum energy, minimum harvesting rate and sum of energy fields are set to the destination values.

Figure 5.4 shows a linear network topology that consists of four nodes. Node S has data to be sent to node D but it hasn't a route to it. So it starts a route discovery process by broadcasting a RREQ message with destination D. As the intermediate node B doesn't have a route to the destination, it updates the RREQ message based on its energy information and rebroadcasts the message. The same processing happens at node C. Once the RREQ reaches D, it replies with a RREP message.

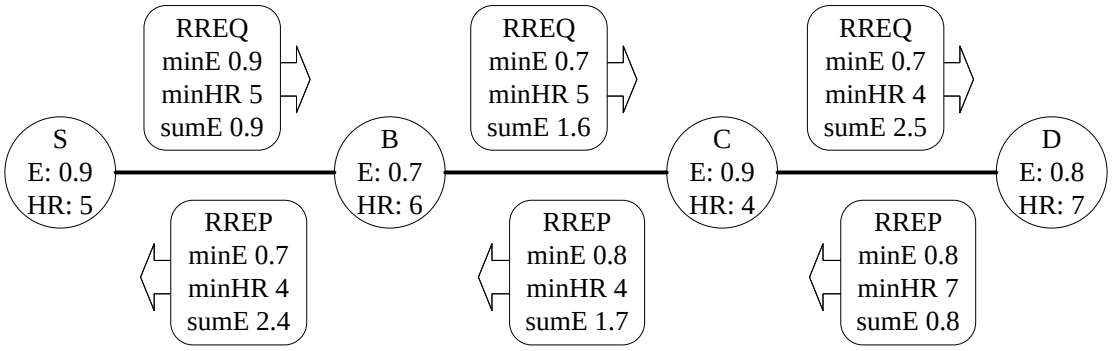


Figure 5.4 Route discovery process in EHAODV.

To simplify the figure, only the energy fields are shown in the RREQ and RREP messages. The reverse and forward routes are built in the source, intermediate and the

destination nodes. After the route discovery process finishes, node S learns that it can reach node D where next-hop is B and path minimum energy equals 0.7, and path sum of energy equals 3.3, and path minimum harvesting rate equals 4. Table 5.1 shows the routing table of node S. it consists of two entries. The first entry is for the neighboring node B. The route entries for the neighboring nodes are automatically created in AODV after exchanging the hello messages. The process of creating the routes to neighboring nodes is called local connectivity. The second route entry is for the destination node D. this entry is created after the route discovery process that is shown in Figure 5.4 took place.

Table 5.1 The routing table of node S of Figure 5.4.

Destination	Next-hop	Hop-count	Min energy	Min harvesting rate	Sum of energy
B	B	1	0.7	5	1.6
D	B	3	0.7	4	3.3

5.2 Route Selection Criteria

Route selection is a basic component of every routing protocol. The route selection is based on metrics including path length, bandwidth, delay, hop count, etc. [41]. AODV protocol is a single path routing protocol i.e. the source creates only one route entry to the destination. AOMDV is a multipath version of AODV [42]. It builds multiple paths during route discovery. When a route is broken, the protocol picks another route from the table.

As mentioned earlier, the energy fields along with the hop-count of the route are used in EHAODV to choose the best route to the destination. The source node can

receive multiple RREP messages from different paths. In AODV, when the source node receives the first RREP, it creates a route based on that RREP. Before it starts sending data and during the route discovery process, if the source node receives a new RREP with lower number of hops, the node updates its routing table with the new path. In EHAODV the selection criteria depends on the route minEnergy, sumEnergy, and minHR in addition to hop-count.

Two energy thresholds are used in EHAODV. The first one is the normal energy threshold (Enormth) which indicates a high level of energy in which the node is capable to operate for enough time before dying. The second threshold is the minimum energy threshold (Eminth) which indicates a low level of energy. When the node energy level becomes less than Eminth, the node stops operating to avoid power outage. A node with available energy less than Eminth is avoided when choosing the route if there is any other route with minE higher than Eminth. The source node never builds a route that doesn't satisfy the condition of minimum energy being higher than Eminth. As shown in Figure 5.5, the source node tries a new route discovery process when the route minimum energy is less than Eminth. The number of route discovery retries is limited to RreqRetries threshold that controls the maximum number of retransmissions of RREQ to discover a route.

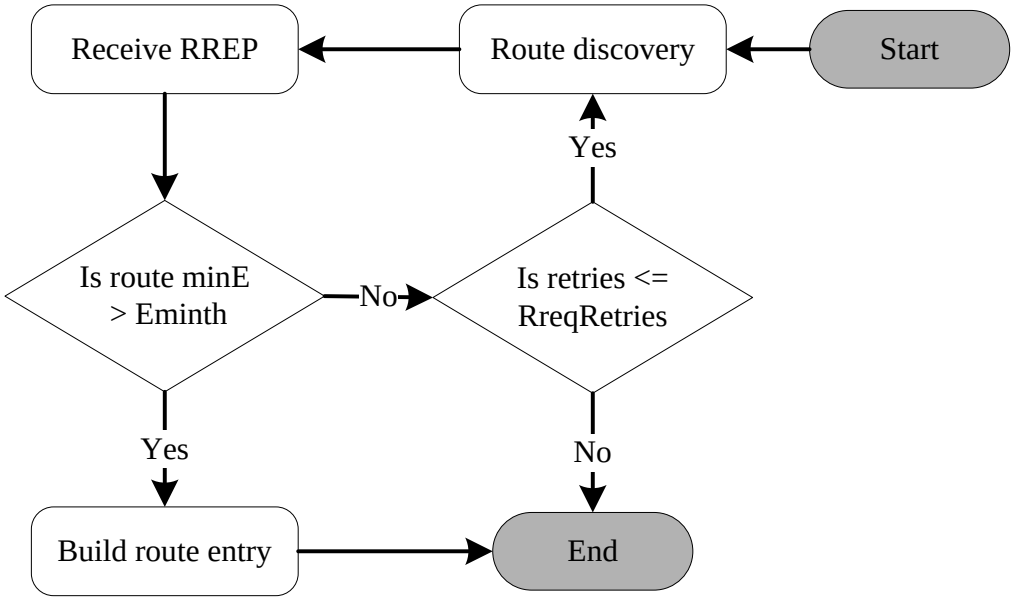


Figure 5.5 Route acceptance criteria in EHAODV.

We have considered two route selection methods. The first method considers the average energy in the path while the second method depends on minimum energy in the path. The following sections explain in details these methods.

5.2.1 Route Average Energy Selection Criteria

In this method, the first priority in route selection is given to the route average energy in the path. The average energy is computed by dividing the total energy of the path (sumEnergy) over the number of hops (hop-count). If the average energy of the newly discovered route is higher than the existing route, the route table is updated by the new route. If the average energy is equal in both routes, the selection depends on the minimum energy harvesting rate (minHR). The route that has higher minHR is chosen. Figure 5.6 shows the route selection process.

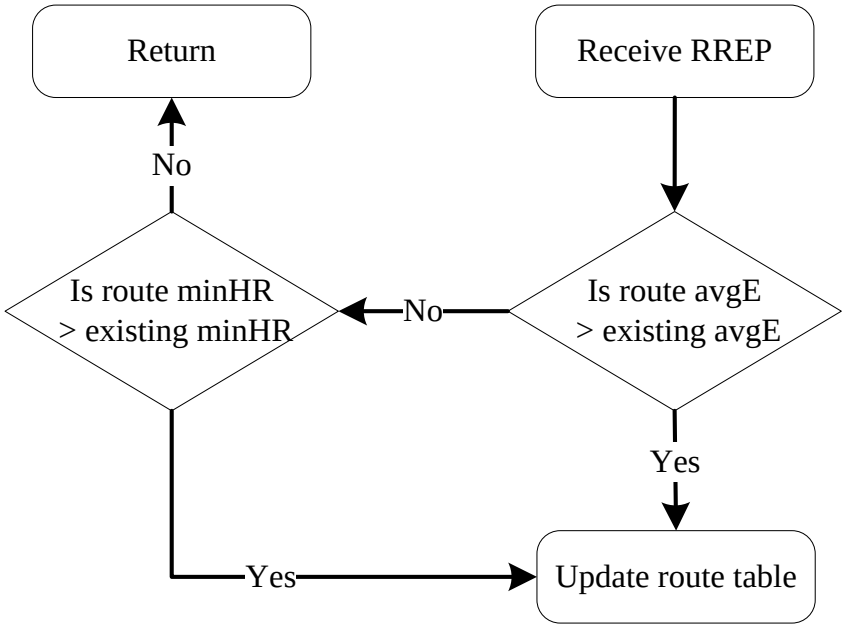


Figure 5.6 Route selection based on average energy and harvesting rate.

5.2.2 Route Minimum Energy Selection Criteria

In this method, when the available energy in each node (minE) is higher than Enormth threshold, the energy is not considered as a constraint in the routing. So the priority in route selection is the hop-count. A route with lower hop-count is preferred. This will achieve minimum delay in normal situation. When the energy level is less than Enormth threshold, the energy is given the first priority in choosing the best route. In this situation, a path with higher number of hop-count and higher minE is preferred over a path with less number of hop-count but with lower minE. Although this strategy will result in higher delay, it is preferred since it saves the energy of weak nodes and gives it more time to harvest energy. Figure 5.7 shows the algorithm of choosing the best route.

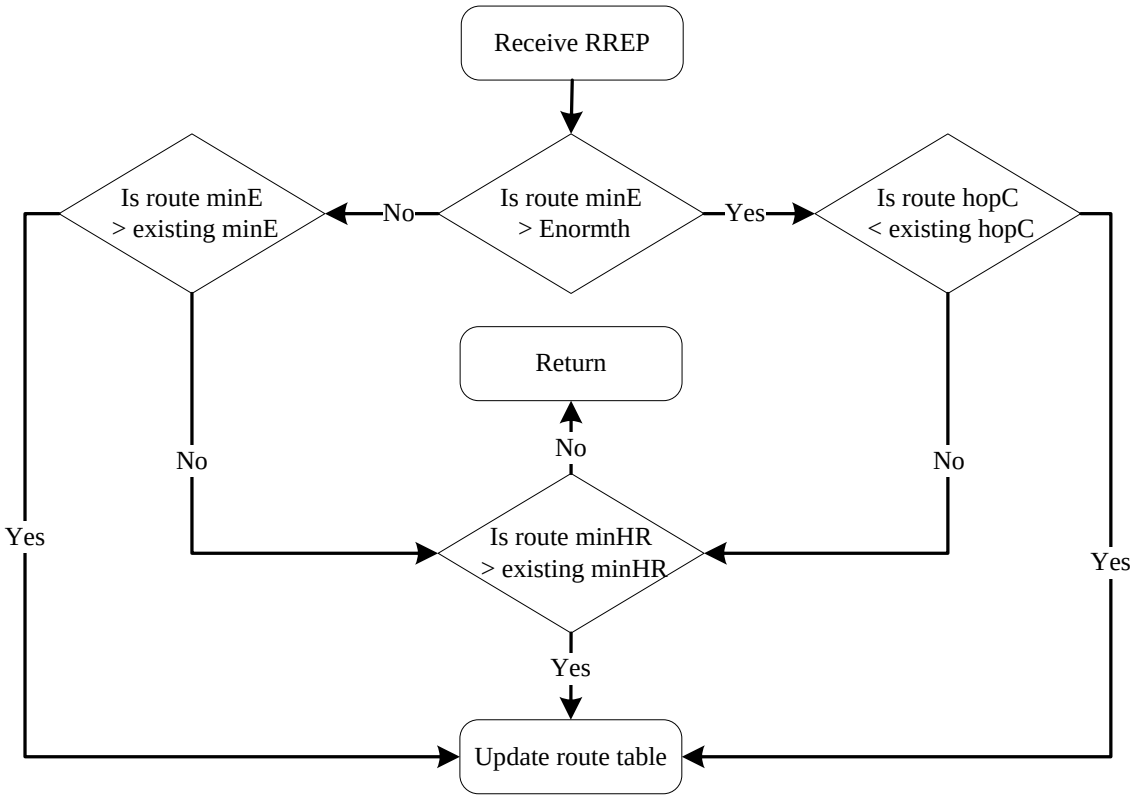


Figure 5.7 Route selection based on minimum energy, hop-count, and energy harvesting rate.

In the case where two available routes have the same hop-count and same minimum energy, the route with highest minHR is chosen.

5.3 Route Repair of EHAODV

It is important to maintain the routes that have been discovered between a pair of nodes. The route is affected with changes that happen for nodes that lie on the same path. Such a path is called an active path [43]. If the source node died during an active session, later it can reinitiate a route discovery process to establish a new route to the destination. In AODV, if the destination or an intermediate node moves and hence the route is broken, a RERR message is sent to the source node [10]. The upstream node which is

closer to the source node lists all the destinations that are not reachable because of the loss of the link in the RERR. The neighboring nodes that receive the RERR message mark the route to destination as invalid and propagate the RERR until it reaches the source node. The source node can reinitiate a route discovery if the route is still needed.

The local connectivity in AODV is managed by broadcasting messages sent by neighboring nodes. When a broadcast message is received from a specific neighbor, the lifetime field associated with that neighbor in the routing table is updated. Each node is required to send a Hello packet every Hello_Interval to inform its neighbors that it is still in the vicinity. The Hello message contains the node's IP and the current sequence number.

A link break might occur during an active communication session. In AODV, this kind of route break causes the source node to create a new route and resend the data again. We propose modifying this behavior such that the node upstream searches for a bypass route to the sink.

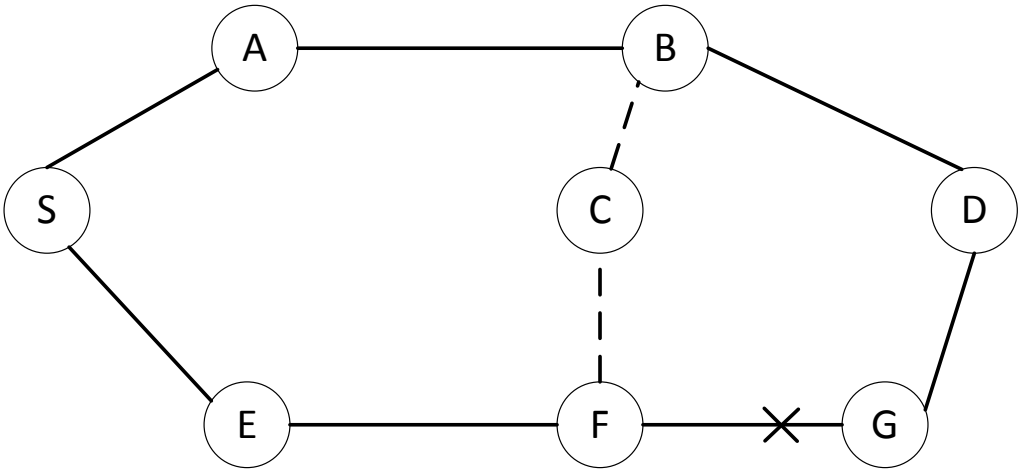


Figure 5.8 Local route repair in EHAODV.

In Figure 5.8, suppose that a route is created through nodes S-E-F-G-D and a data session is running. During that and before the data session is finished, the energy of node G was drained and hence F-G link is broken, F realizes a link break in the route during active session so it buffers the incoming packets from S and sends RERR message to S to notify it to extend the route lifetime. Node F initiates a local route repair through its neighboring nodes. For example, the bypass route F-C-B-D is built and used to resume the data session between S and D. F sends the buffered packets to D through the newly discovered route. If the node upstream fails in repairing the broken link during the route active timeout interval, the source node reinitiates a route discovery to send the data again.

The same format of RERR message of AODV was used. The *No_delete_flag (N)* is exploited to implement the route repair mechanism. In EHAODV, the intermediate node sets the flag to true such that the upstream nodes don't invalidate the unreachable routes in the RERR. When a node receives a RERR with flag N set to true, it resets the route active timeout to its initial value.

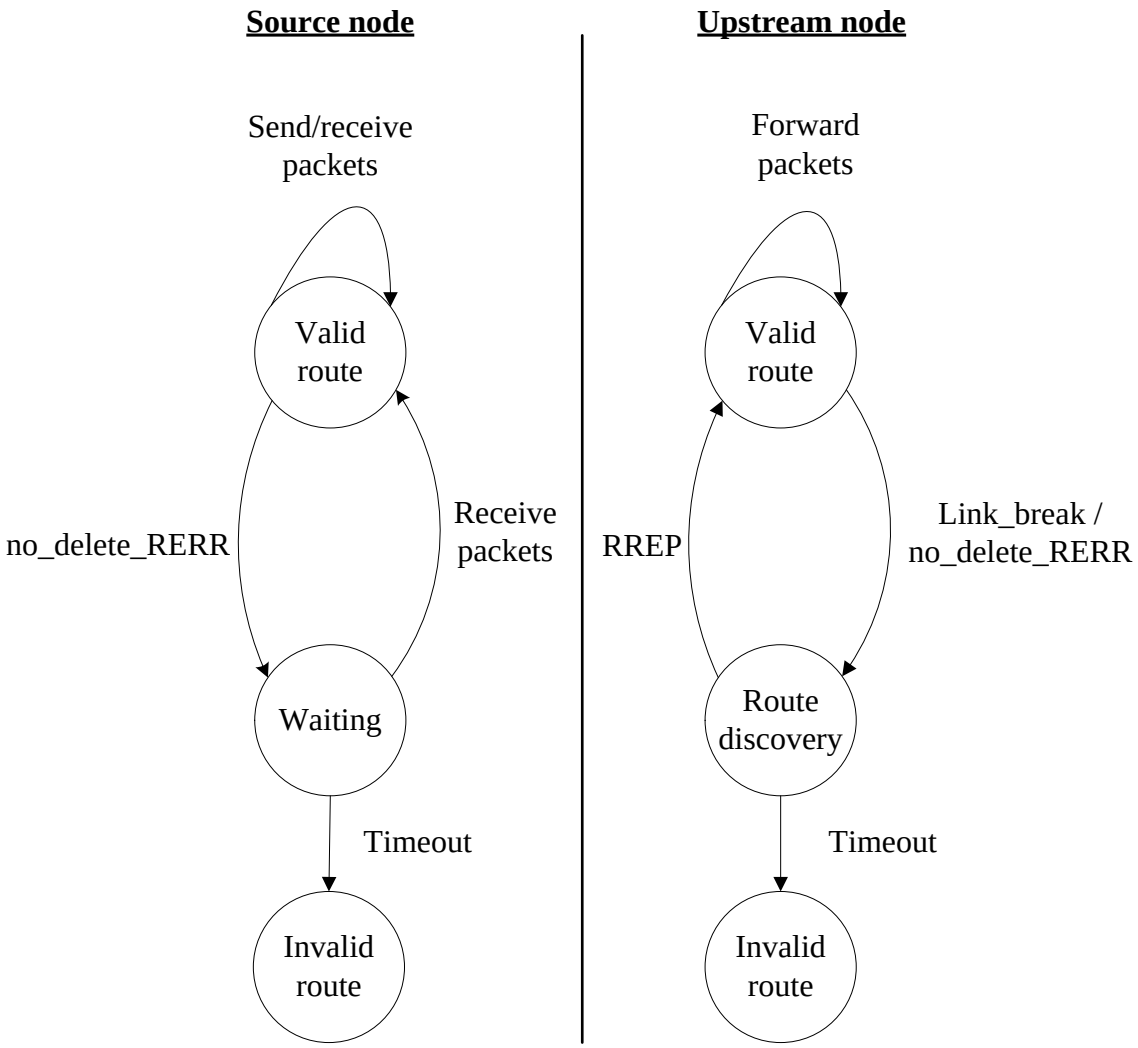


Figure 5.9 Route state diagram of the source and upstream nodes during route repair process.

Figure 5.9 shows the states of the route upstream node (right) and the source node (left). Initially, a route discovery process is done by the source node and hence an active route is created in the source node and the intermediate nodes. The route has a timeout interval in which the route is deleted if it is not used. Whenever a packet is sent through the route, the timeout interval is set to its initial value. When a link break happens, the upstream node sends a RERR message to the source with *no_delete_flag* is set to true and starts a route discovery process to the destination. When the source node receives the RERR, the

route timeout value is set to the initial value and the node enters a waiting state. If the route discovery at the node upstream succeeds, the node resumes sending the queued packets to the destination; otherwise, the route is marked as invalid. When the source node receives ACK packet from the destination, the source resume sending packets to the destination. Otherwise, if timeout is triggered and no packet is received from the upstream node, the source node is forced to invalidate the route. The timeout value in the route repair mechanism is the same *active_route_timeout* that is used in EHAODV.

CHAPTER 6

SIMULATION AND RESULTS

EHAODV protocol was implemented using ns-3 simulator. In this chapter, we introduce different simulation scenarios that show the performance of EHAODV versus the original AODV. The performance metrics that used were discussed in section 4.2.

6.1 Simulation Setup

A grid topology consists of 9X11 Wifi nodes was built as shown in Figure 6.1.

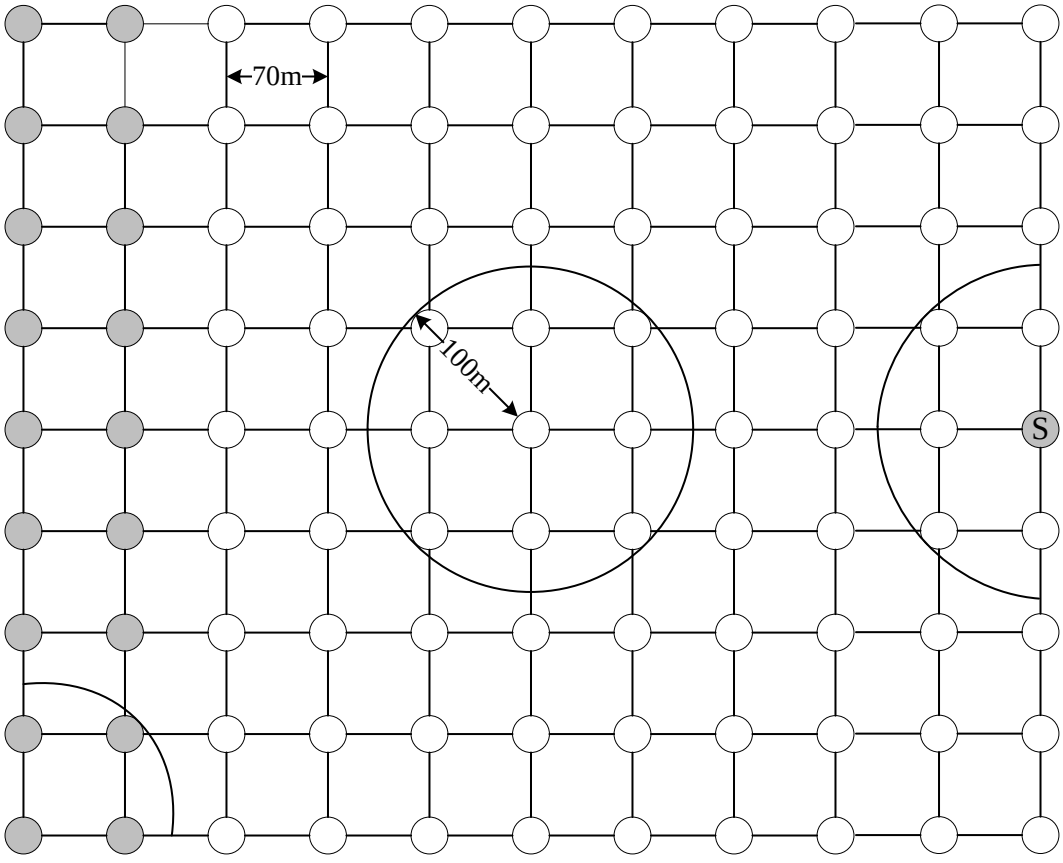


Figure 6.1 Nodes placement.

The transmission distance of the node is 100 meters while the node displacement is 70 meters. The outdoor transmission distance of MICAz mote is 75m to 100m [44]. A node located at the center can communicate directly with the surrounding eight nodes. The topology is fixed (i.e. the nodes don't move from their initial location). One sink application is installed on the gray node at the right. The first left two columns of the grid (gray) are the source nodes. Each one of the source nodes has a file to be sent to the sink node. The source nodes start sending their data to the sink at the beginning of the simulation and stop when finish sending their file. Eighteen concurrent flows are seen at the sink node. The rest of the nodes are used as relay nodes to connect the sources with the sink.

The AODV protocol has many parameters that affect the routing process. These parameters include timeout, buffer size, threshold, etc. as shown in Table 6.1. We use the same parameters for EHAODV.

Table 6.1 AODV and EHAODV protocol parameters [10].

Parameter	Description	Value
HelloInterval	HELLO messages emission interval	1 sec
RreqRetries	Maximum number of retransmissions of RREQ to discover a route	2
RreqRateLimit	Maximum number of RREQ per second	10
RerrRateLimit	Maximum number of RERR per second	10
NodeTraversalTime	Conservative estimate of the average one hop traversal time for packets and should include queuing delays, interrupt processing times and transfer times	40 millisec
NextHopWait	Period of our waiting for the neighbour's RREP_ACK = 10 ms + NodeTraversalTime	50 millisec
ActiveRouteTimeout	Period of time during which the route is considered to be valid	3 sec
MyRouteTimeout	Value of lifetime field in RREP generating by this node = 2 * max(ActiveRouteTimeout, PathDiscoveryTime)	11.2 sec
BlackListTimeout	Time for which the node is put into the blacklist = RreqRetries * NetTraversalTime	5.6 sec
NetDiameter	Net diameter measures the maximum possible number of hops between two nodes in the network	35
NetTraversalTime	Estimate of the average net traversal time = 2 * NodeTraversalTime * NetDiameter	2.8 sec
PathDiscoveryTime	Estimate of maximum time needed to find route in network = 2 * NetTraversalTime	5.6 sec
MaxQueueLen	Maximum number of packets that we allow a routing protocol to buffer	64
MaxQueueTime	Maximum time packets can be queued	30 sec
AllowedHelloLoss	Number of hello messages which may be loss for valid link	3
GratuitousReply	Indicates whether a gratuitous RREP should be unicast to the node originated route discovery	True
DestinationOnly	Indicates only the destination may respond to this RREQ	False
EnableHello	Indicates whether a hello messages enable	True
EnableBroadcast	Indicates whether a broadcast data packets forwarding enable	True

6.1.1 Energy Model

The energy resource is the major issue in EHAODV. An energy model and energy harvesting model were installed on the nodes. The energy models of ns-3 were discussed in section 2.2. The nodes in the simulation start with initial amount of energy to initiate the communication between the nodes. Whenever a node transmits or receives data, it consumes energy according to the energy model parameters. The parameters that affect the amount of energy consumption are listed below. The values are compatible with MICAz mote [44].

- Voltage: the circuit voltage is a configurable parameter in ns-3. The default voltage value in ns-3 is 3V. The default value was used in the simulation.
- The transmitter current: the default current in ns-3 is 17.4 mA.
- The receiver current: the default current in ns-3 is 19.7 mA.
- The transmission time: this depends on the channel bandwidth. The channel type that used is 6Mbps OFDM.

The energy model has other parameters as idle, switching, and sleeping current. In the simulation, Tx and Rx currents were considered while other currents like idle, sleep, and switching are set to zero. The amount of energy consumption is calculated as Eq. 6.1[45].

$$Energy = Voltage * Current * Time \quad (6.1)$$

where the time is computed as in Eq. 6.2 [46].

$$Time = File\ size\ /\ Bandwidth \quad (6.2)$$

The sink node is assumed to be powerful enough such that it never turns off.

6.1.2 Energy Harvesting Model

The energy harvesting model of ns-3 is described in section 2.2. The energy harvesting model is installed on all nodes in the simulation. The amount of the harvested energy is controlled by the minimum harvesting power and maximum harvested power parameters. The minimum harvesting power is set to zero for all nodes, while the maximum harvesting power is different. This simulates different energy harvesting capabilities of the nodes based on the availability of the ambient energy (e.g. node in the shade or under direct sun light). The nodes that have high maximum harvesting power harvests more energy than the nodes that has lower harvesting power. The maximum harvesting power of the nodes is modeled as a uniform random number between 2mW to 8mW. The node harvests an amount of energy every second where this amount is a uniformly distributed value between the minimum harvesting power and the maximum harvesting power of the node. The amount of harvested energy for a node in the entire simulation is computed as in Eq. 6.3:

$$Node\ Tot.Harv.Energy = Sim.Time * Avg(minhp, maxhp) \quad (6.3)$$

Figure 6.2 demonstrates as snap shot of total harvested energy of the nodes for a test of 800 sec. The X-axis and Y-axis numbers indicates the node index in the network topology. The *minhp* is set 0 and the *maxhp* of the nodes is chosen randomly between 2 mW to 8 mW. The average harvesting power at any node is 2.5 mW. Then, as in Eq. 6.3, every node will harvest on average as high as 2J

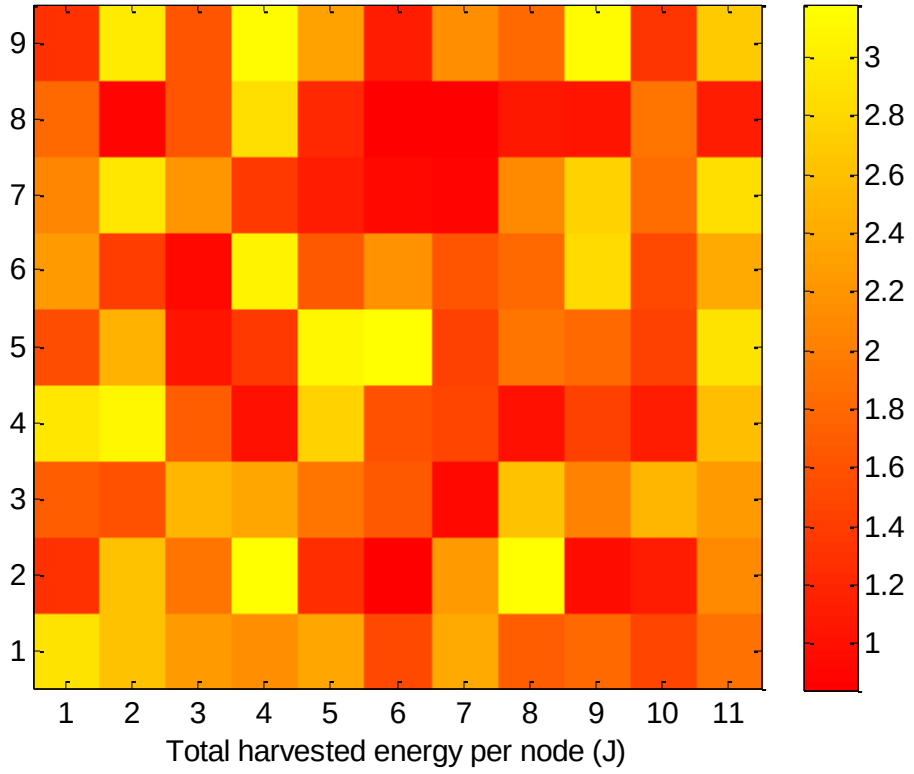


Figure 6.2 Total harvested energy per node.

The cells with dark color indicates low amount of energy around one joule. On the other hand, cells with light color indicates high amount of energy around three joules. The amount of harvested energy of a node follows the harvesting rate value.

6.2 Simulation Scenarios

In the following sections, we present different simulation scenarios that we have conducted to study the performance of EHAODV against the original AODV. For each scenario, we show the simulation parameters, results, and discussion. In section 5.2, we mentioned that two route selection criterions were implemented in EHAODV. In the following sections, EHAODV using the route average energy selection criteria is denoted as EHAODV-a while EHAODV using the route minimum energy selection criteria is denoted as EHAODV-m. Moreover, EHAODV with route repair is denoted as EHAODV-r.

6.2.1 Studying the Effect of Varying Number of Hops

In this scenario, a simulation experiment is held where the variable is the number of hops between the source nodes to the sink node. The number of hops represents the direct distance from the source nodes to the sink in terms of hops. The goal of the scenario is to show the performance of EHAODV-a, EHAODV-m, and original AODV in terms of end-end delay, goodput, and packet loss ratio when changing the number of hops from the source nodes to sink. The performance metrics were discussed in section 4.2. The simulation parameters are shown in Table 6.2.

Table 6.2 Simulation Parameters.

Parameter	Value
Test Area	560m X 700m
Number of nodes	99
Placement	9 X 11 grid
Radio range	100 m
Nodes separation	70 m
Transmission bandwidth	6 Mbps
Application traffic	On time: 5 sec, Off time: 3.1 sec Rate: 13 Kbps, Transport: TCP
Number of data sources	18
Packet size	536 Bytes
Number of packets per source	400 packets
Source to destination physical distance	Varying 6 to 10 hops
Energy harvesting	Minhp: 0, maxhp: [2 mW – 8 mW]
Node off time (after power outage)	50 sec
Routing protocol	AODV, EHAODV-a, EHAODV-m
Simulation time	800 sec
Scenario replications	30

The network topology, sources, and sink are shown in Figure 6.1. The application which is installed on the source nodes sends data packets to the sink in on/off manner. The application of all nodes starts sending data at time 10s. As soon as the source has packets to be sent to the sink, the application sends data for five seconds at rate of 13 Kbps then stops sending for three seconds and so on. This process continues until the source finishes sending the total number of packets which is 400 in this scenario. The scenario is replicated 30 times to achieve 95% level of confidence.

The effect of physical distance between a source to a destination is studied in this scenario. The route hop-count can be higher than the physical distance based on the path that is constructed during the route discovery process. The physical distance is changed by moving the sink node left and right. The physical distance in Figure 6.1 is ten hops. In order to get physical distance of nine hops, the sink application is installed on the node which is one step to the left and so on. When a node is turned off because of low energy, it waits for about 50 seconds in order to recharge and have enough energy and then it turns on.

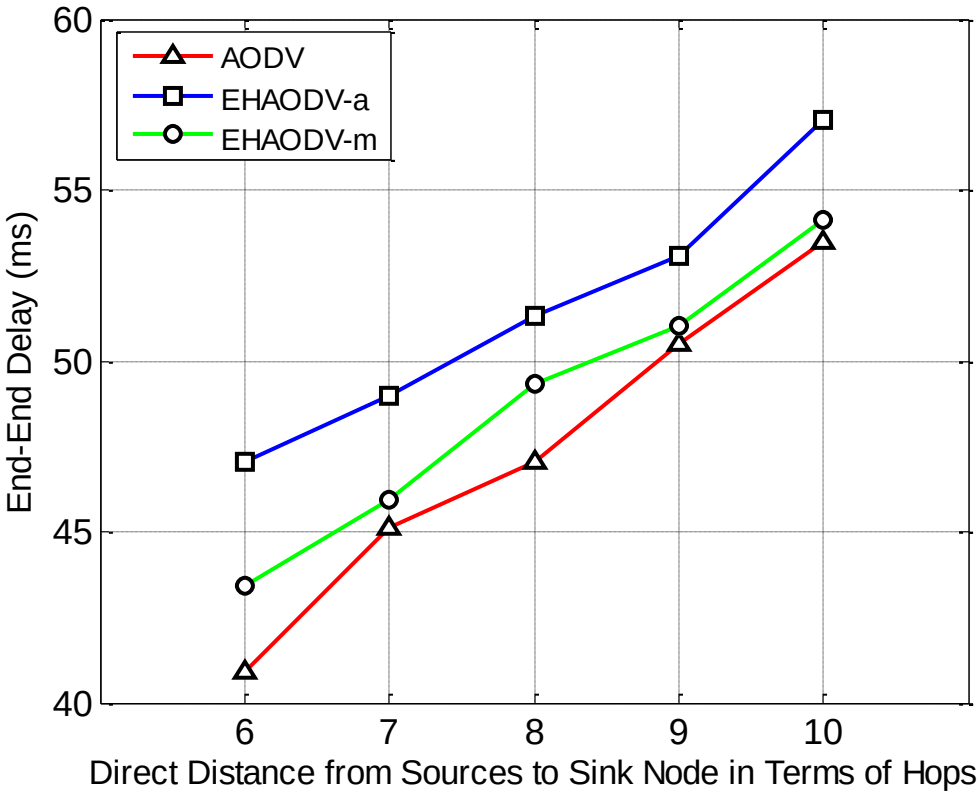


Figure 6.3 Average packet end-end delay per flow. Eighteen TCP flows and file size of 209Kbytes per flow.

Figure 6.3 shows the average packet end-to-end delay versus the number of physical hops. It is expected that the value of end-to-end delay increases when the number of hops is increased. The end-to-end delay is affected by the path length and the number of nodes in the path. Longer path increases the propagation delay while higher number of nodes increases the queuing and processing delay. AODV has the lowest end-to-end delay. This is explained by the type of selection criteria that each protocol has. AODV selects paths with lowest number of hops regardless of energy in each node. EHAODV-a has the highest end-to-end delay since it accepts paths with higher hop-count. EHAODV-m has end-to-end delay in between the other protocols but closer to AODV. In order to quantify the performance of the protocols, a two-factor full factorial design experimental design

analysis is done to determine the effects of the factors on the delay [47]. Table 6.3 shows the effects of the routing protocol and number of hops factors on the delay.

Table 6.3 Computation of effects of the protocols on end-end delay.

Hop-count	AODV	EHAODV-a	EHAODV-m	Row Sum	Row Mean	Row Effect
6	40.91	47.07	43.43	131.41	43.80	-5.42
7	45.14	48.99	45.94	140.07	46.69	-2.54
8	47.03	51.32	49.35	147.70	49.23	0.0049
9	50.50	53.06	51.05	154.61	51.53	2.307
10	53.49	57.05	54.10	164.65	54.88	5.654
Column Sum	237.08	257.50	243.88			
Column Mean	47.41	51.50	48.77		49.23	
Column Effect	-1.81	2.26	-0.4544			

For each row (or column), we compute the mean of delay in that row (or column). Overall means are also computed. The difference between a row (or column) mean and overall mean gives the row (or column) effect. The results of the analysis are interpreted as follows. The packet end-to-end delay is 49.23 milliseconds with an average protocol and average hop-count.

The delay with AODV is 1.81 milliseconds lower than that with an average protocol, and the delay with EHAODV-m is 0.45 milliseconds lower than that with an average protocol. The delay with EHAODV-a is 2.26 milliseconds higher than the average. This is equivalent to saying that the mean difference between AODV and EHAODV-m is 1.36

millisecond. Similarly, the mean difference between AODV and EHAODV-a is 4.07 milliseconds.

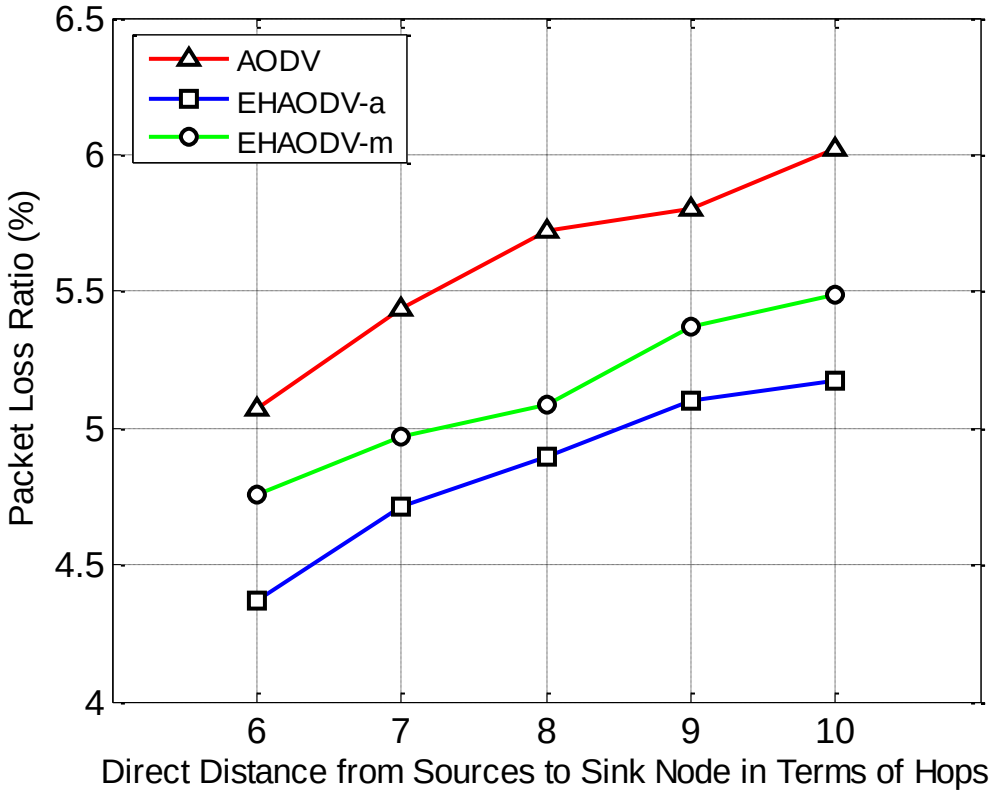


Figure 6.4 Average packet loss ratio per flow. Eighteen TCP flows at average rate of 8Kbps and file size of 209Kbytes per flow.

Figure 6.4 shows the average packet loss ratio versus the number of physical hops. Packet loss is caused by two reasons. The first reason is the communication interference, which is anything which alters, modifies, or disrupts a message as it travels along the channel [48]. The second reason is the node's power outage. If a node is participating in an active communication flow and then dies, this causes an interruption to the flow and consequently loss for the packets that are sent by the source but not yet received by the sink. On the other hand, higher number of hops causes packet loss ratio to increase since

the probability of packet loss in longer path is higher than shorter paths. EHAODV-a achieved the lowest packet loss ratio. While the original AODV achieved highest packet loss ratio.

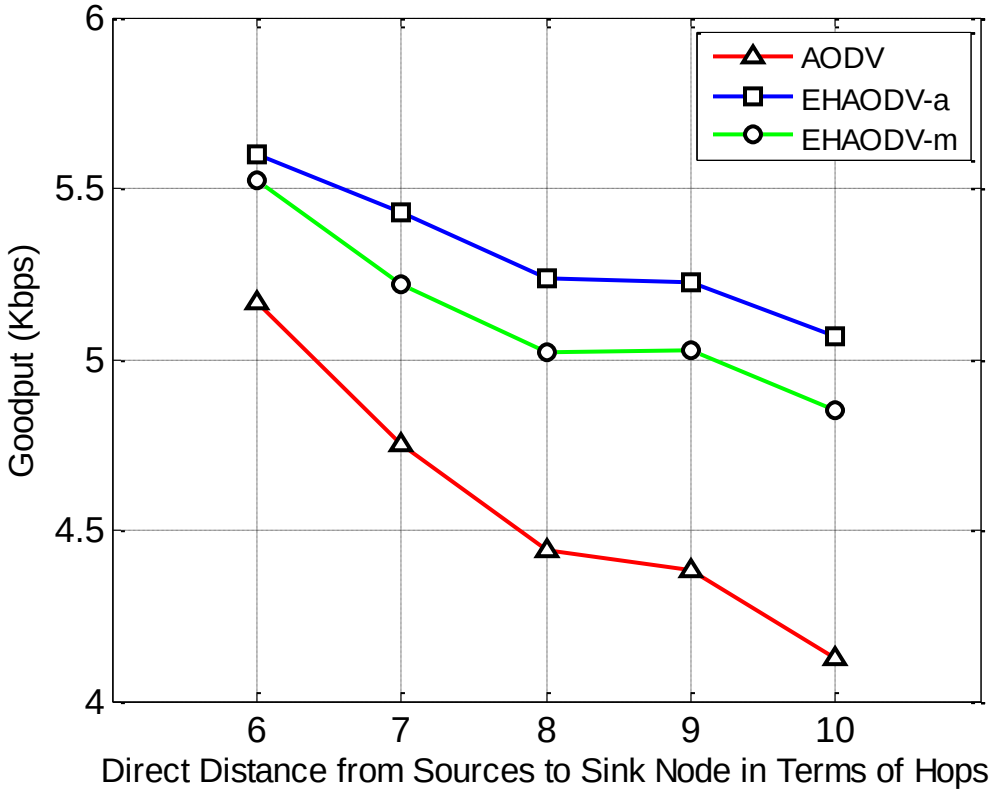


Figure 6.5 Average goodput per flow. Eighteen TCP flows at average rate of 8Kbps and file size of 209Kbytes per flow.

Figure 6.5 shows the average flow goodput versus number of hops. The goodput decreases when the hop-count increases. EHAODV-a achieved the highest goodput while AODV achieved the lowest. EHAODV-m performed in between the other two protocols but closer to EHAODV-a. The enhancement in goodput of EHAODV is higher when the hop-count increases. This result can be attributed to the fact that when the path from the

source to destination is long, it is more probable to find a better route in terms of energy than when the route is short.

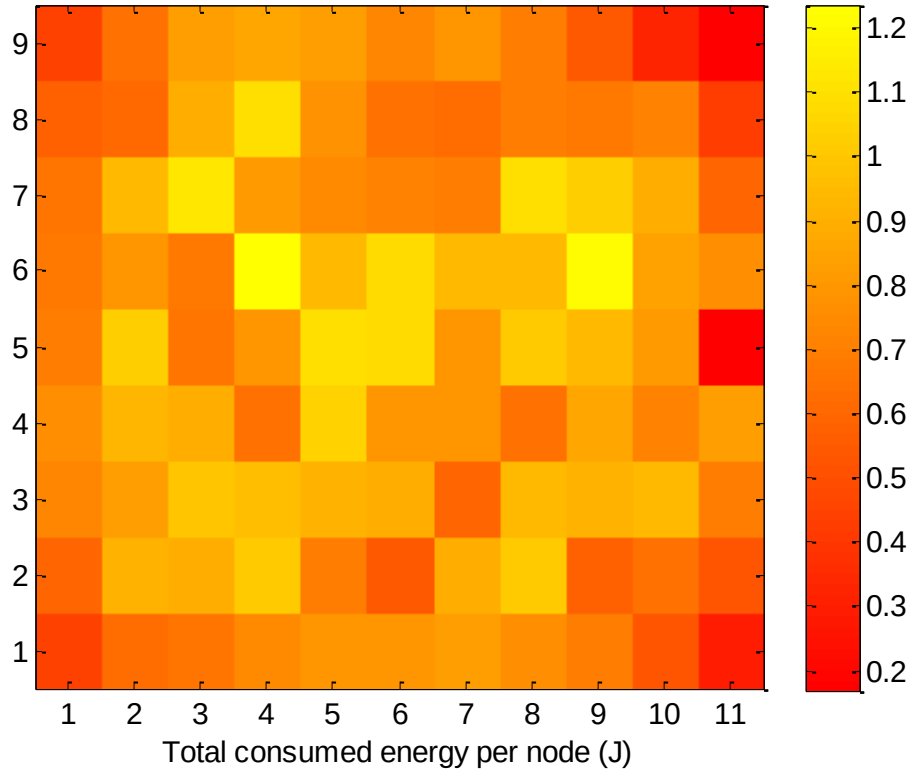


Figure 6.6 Total consumed energy per node.

Figure 6.6 shows a snapshot of the total consumed energy of each node. The range of consumed energy is from 0.2 to 1.2 J. The nodes that at the middle of the topology consumed very high amount of energy. This is because the intermediate nodes forward the packets toward the sink. The nodes at the corners and the edges of the topology consumed less energy. It is unlikely that the nodes at the corners participate in the communication. The node at the middle of the column at the right is the sink. As expected, it consumed a low amount of energy since its energy model was configured to consume low energy.

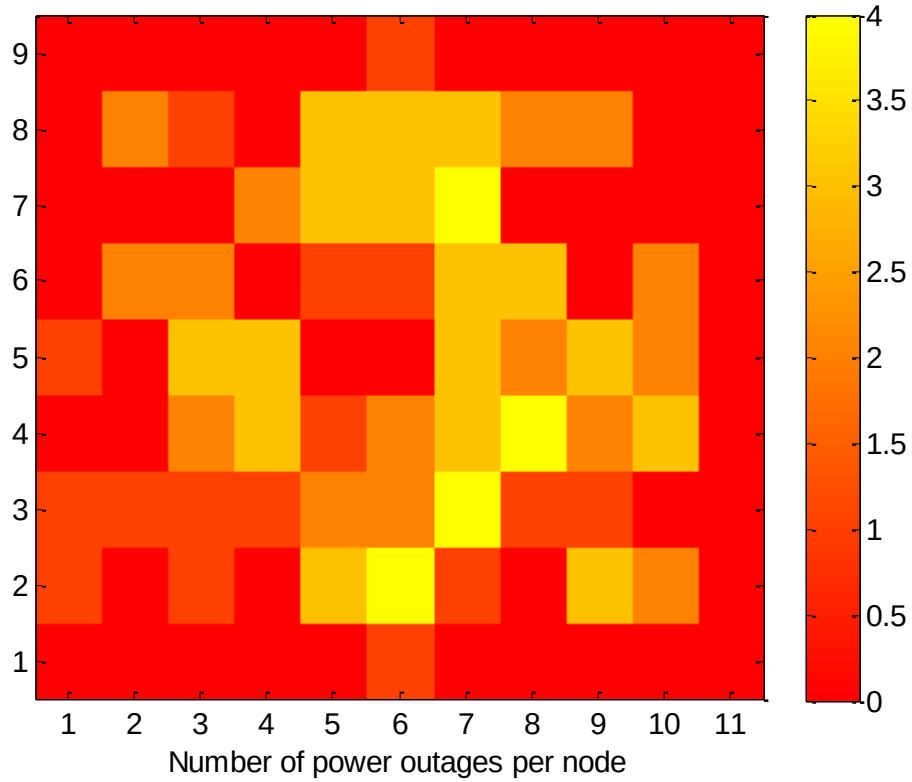


Figure 6.7 Total power outages per node.

Figure 6.7 shows a snapshot of the total power outages of each node. The highest count of outages is four while many nodes are never turned off. The number of power outages of the node depends on its activity which is reflected in the total consumed energy, and the number of power outages. The weak nodes that are located in the middle of the topology have high count of power outages. Since it is unlikely that the nodes at the corners participate in the communication, their energy consumption is low and hence they were never turned off.

Table 6.4 shows the effects of the protocols on the different performance metrics. The percentage of the enhancement or degradation with respect to AODV is shown. Each value in the table represents the average performance of the protocol with the

corresponding metric. The average performance metric for each protocol is computed by summing up the metric values at all hops points and dividing the result over the total number of points which is five in this scenario. For example, the goodput with EHAODV-a is on average 16.15% higher than AODV. Similarly, the packet loss ratio with EHAODV-m is on average 8.49% lower than AODV.

Table 6.4 Computation of effects of protocols on the performance with respect to AODV. (+) and (-) signs indicate increment or decrement in the corresponding performance metric, respectively.

Performance metric	Protocol	
	EHAODV-a (%)	EHAODV-m (%)
End to end delay	8.61	2.87
Packet loss ratio	-13.55	-8.49
Goodput	16.15	12.17

6.2.2 Studying the Effect of Fixed Harvesting Rate and Energy Based Threshold

In this scenario we study the effect of fixed energy harvesting rate versus the variable energy harvesting rate in the network. In addition, we study the effect the energy based threshold and time based threshold. The nodes in the previous scenario have different energy harvesting capabilities. In this scenario the nodes are configured to have similar energy harvesting rate. *minhp* is set to 0 while *maxhp* is set to 5 mW for all nodes. The average energy-harvesting rate in this scenario is equivalent to scenario in section 6.2.1 but the nodes in this scenario have similar harvesting capabilities.

In the previous scenario, when a node is turned off, it waits about 50 seconds then it is turned on. We call this method as time based threshold. In this scenario we test the effect

of the threshold by using different types of threshold. When a node is turned off, it stays off until its remaining energy reaches 100 mJ then it is turned on. We call this method as energy based threshold (ebth). The two variants of configurations which are fixed harvesting rate and energy based threshold, are compared with EHAODV-m in the previous scenario. Table 6.5 shows the simulation parameters of this scenario. The other parameters are the same as in Table 6.2.

Table 6.5 Simulation parameters.

Parameter	**Value**
Routing protocol	EHAODV-m, EHAODV-fhr, EHAODV-ebth
Energy harvesting	minhp: 0, maxhp: 5 mW
Energy rise threshold	100 mW

EHAODV-fhr (EHAODV with fixed energy harvesting rate) has the same configurations as EHAODV-m except energy harvesting rate. EHAODV-ebth has the same configurations as EHAODV-m but the node rise threshold is based on energy.

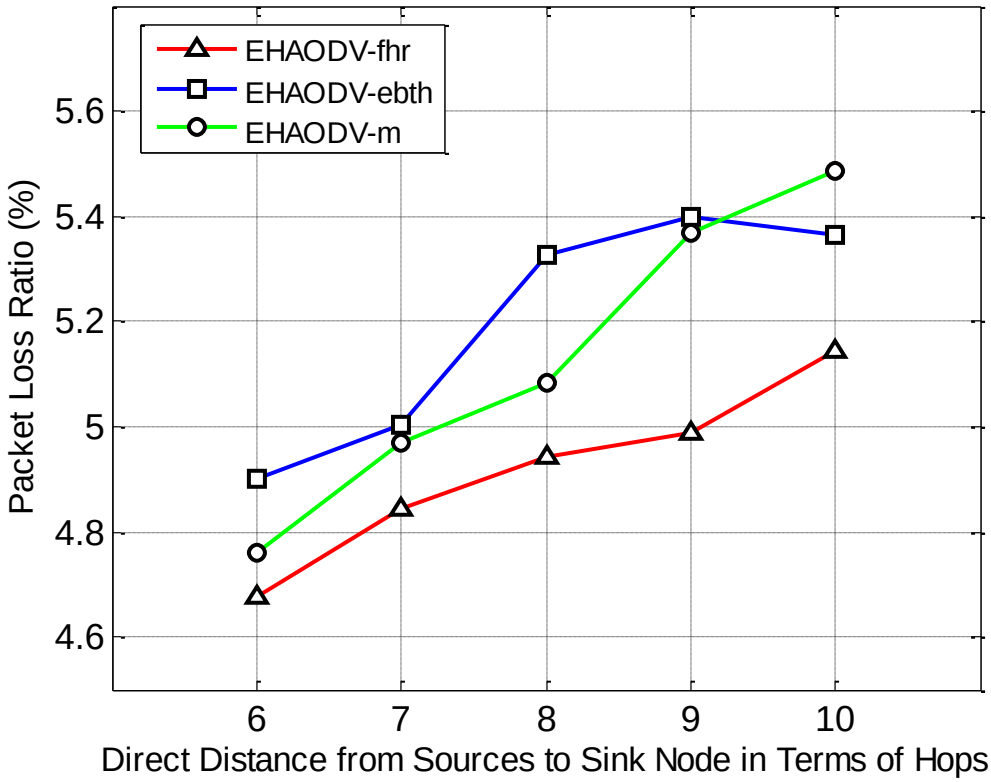


Figure 6.8 Average packet loss ratio per flow. Eighteen TCP flows at average rate of 8Kbps and file size of 209Kbytes per flow.

Figure 6.8 shows the average packet loss ratio versus number of hops. EHAODV with fixed energy harvesting rate achieved the lowest loss ratio.

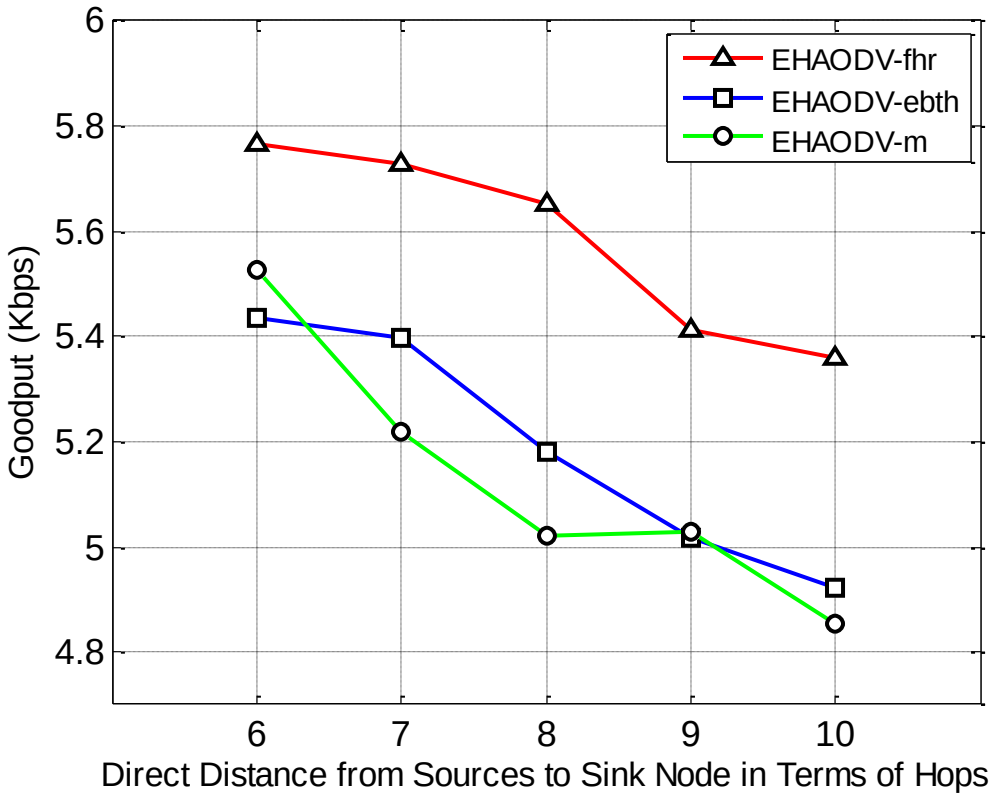


Figure 6.9 Average goodput per flow. Eighteen TCP flows at average rate of 8Kbps and file size of 209Kbytes per flow.

Figure 6.9 shows the average goodput versus the number of hops. EHAODV with fixed energy harvesting rate achieved the highest rate while EHAODV-ebth gave a close goodput to EHAODV-m.

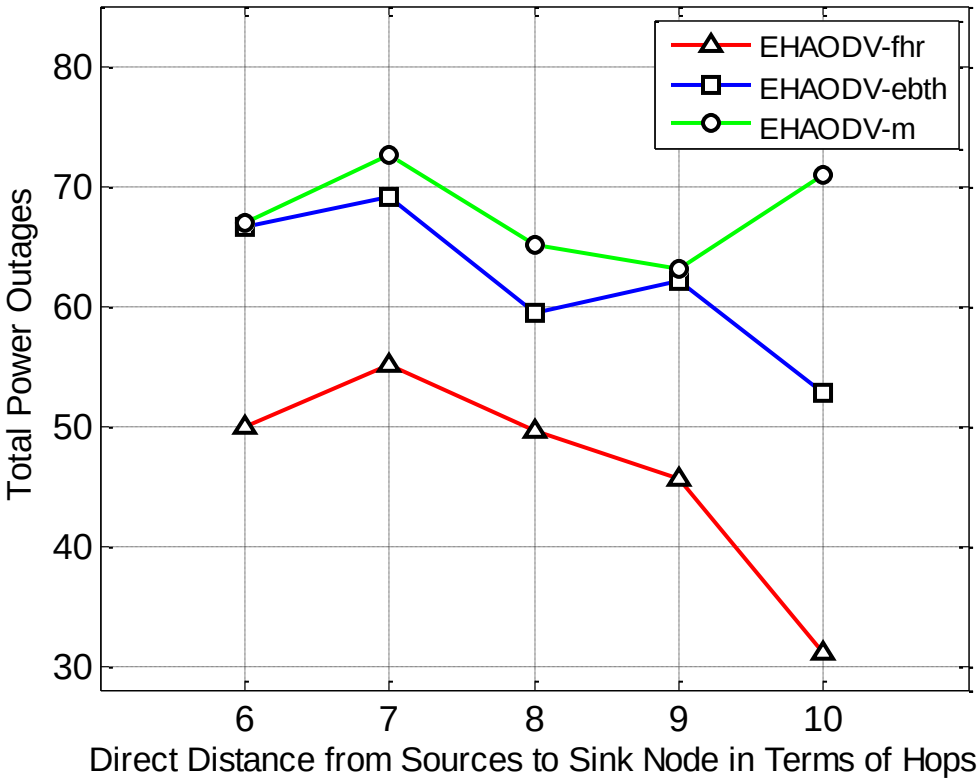


Figure 6.10 Total power outages. Eighteen TCP flows at average rate of 8Kbps and file size of 209Kbytes per flow.

Figure 6.10 shows the average of total power outages of each scenario versus the number of hops. EHAODV-fhr has the lowest number of outages. EHAODV-m and EHAODV-ebth behave similarly in terms of outages except at number of hops of ten. The nodes in EHAODV-fhr scenario are with similar harvesting capabilities so there are no weak nodes that will die frequently. While in the other configurations, some of the nodes have low harvesting rate so they are turned on and off frequently. This explains why EHAODV-fhr outperformed other protocols in packet loss ratio and goodput. The energy based and time based threshold didn't show a significant difference in the performance.

Table 6.6 shows the effects of the protocols on the different performance metrics. The percentage of the enhancement or degradation with respect to EHAODV-m is shown.

Table 6.6 Computation of effects of protocols on the performance with respect to EHAODV-m. (+) and (-) signs indicate increment or decrement in the corresponding performance metric, respectively.

Performance metric	Protocol	
	EHAODV-fhr (%)	EHAODV-ebth (%)
Packet loss ratio	-4.18	1.27
Goodput	8.84	1.19
Power outage	-31.77	-8.53

6.2.3 Studying the Effect of Varying Energy Harvesting Rate

In this scenario, a simulation experiment is conducted where the variable is the number of hops between the source nodes to the sink node and the energy harvesting rate (EHR). The goal of this scenario is to study the effect of varying energy harvesting rate on the network performance. EHAODV-m protocol is used in this scenario. The simulation parameters are listed in Table 6.7.

Table 6.7 Simulation parameters.

Parameter	Value
Routing protocol	EHAODV-m
Energy harvesting	minhp: 0 maxhp: [2-8], [4-10], [6-12] mW
Source to destination physical distance	Varying 6 to 10 hops

The other simulation parameters of this scenario are the same as in Table 6.2

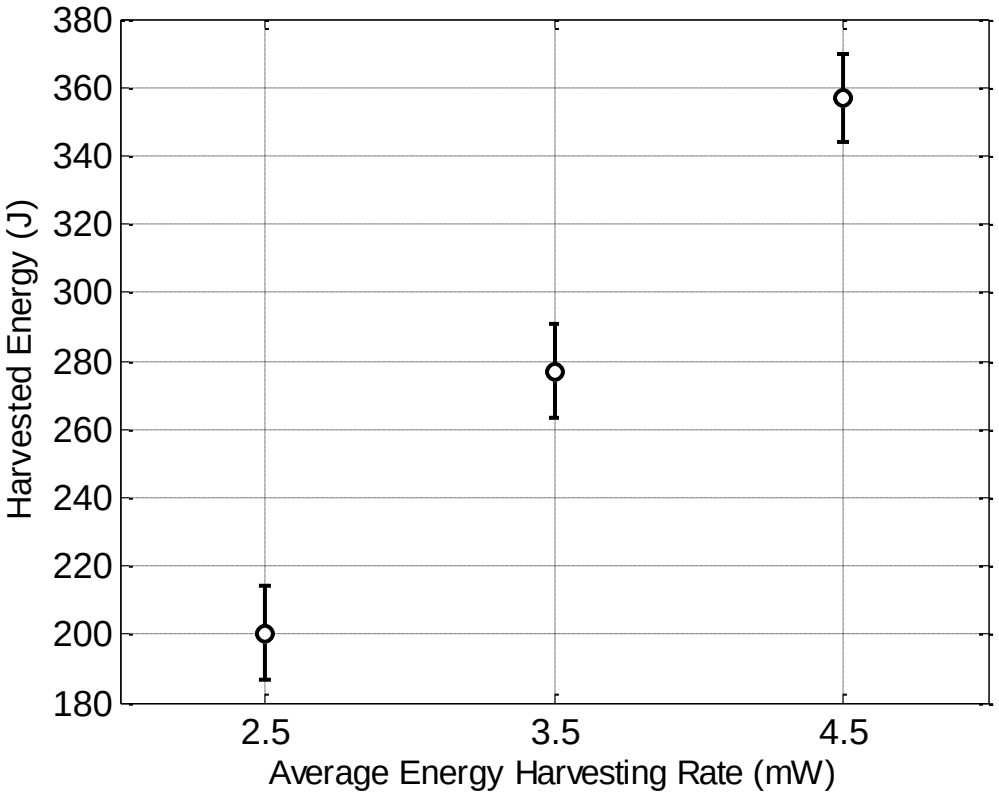


Figure 6.11 Average harvested energy in the network.

Figure 6.11 shows the average harvested energy in the network versus the energy harvesting rate at 95% confidence level. The average harvested energy can be calculated as Eq. 6.3.

Figure 6.12 shows the average packet end-to-end delay of the flows versus the number of hops for different energy harvesting configurations. The higher energy harvesting rate configuration achieved lower end-to-end delay. When *maxhp* is configured with higher value the node's power outages is less. End-to-end delay is directly affected by the number of hops from the source to destination.

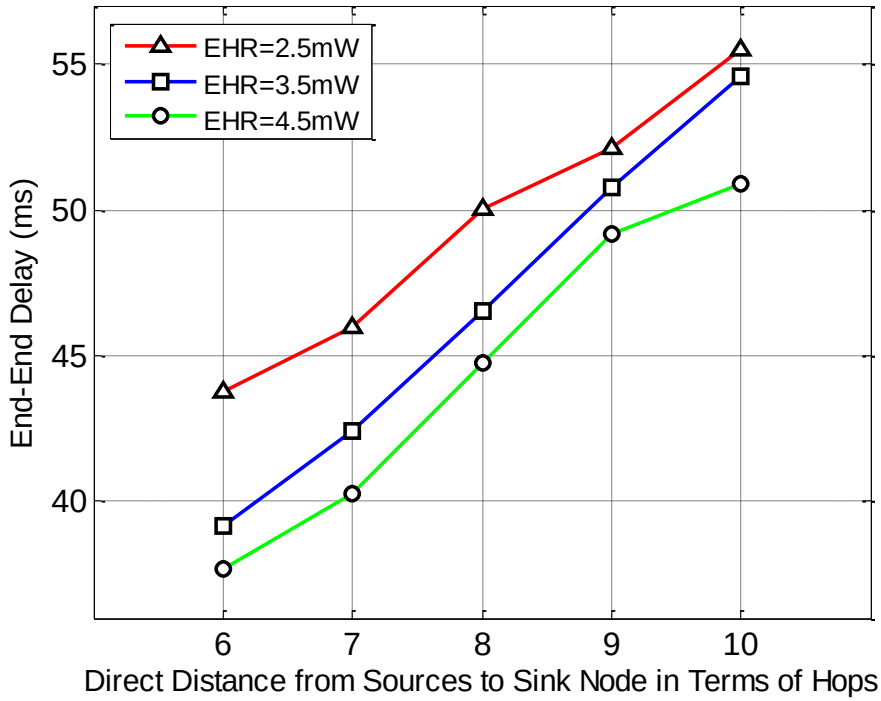


Figure 6.12 Average packet end-end delay per flow. Eighteen TCP flows at average rate of 8Kbps and file size of 209Kbytes per flow.

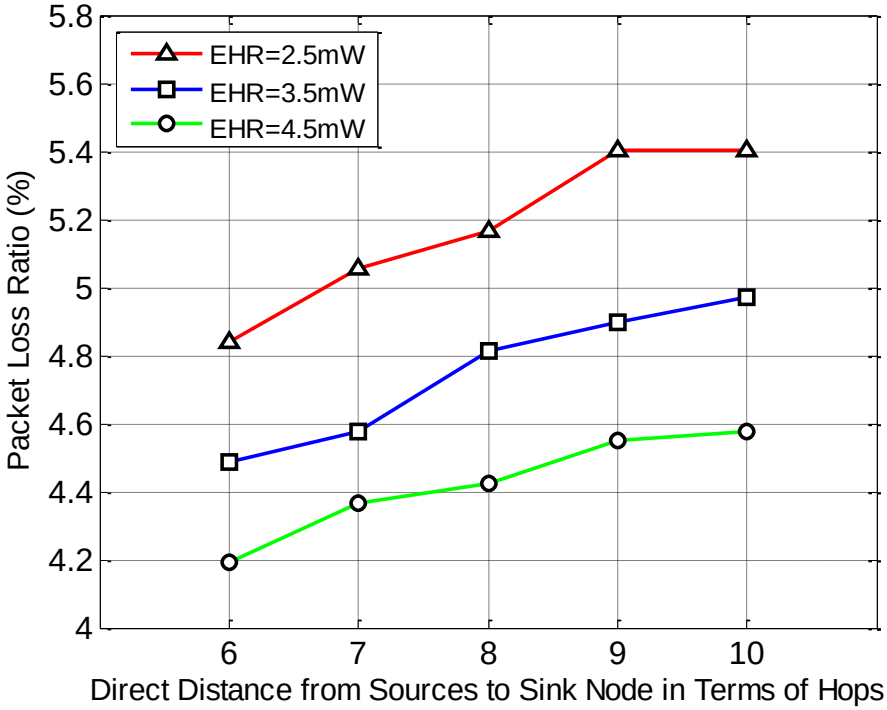


Figure 6.13 Average packet loss ratio per flow. Eighteen TCP flows at average rate of 8Kbps and file size of 209Kbytes per flow.

Figure 6.13 shows the average packet loss ratio of the TCP flows versus the number of hops. Packet loss ratio increases as the number of hops increases. As expected, the configuration with highest energy harvesting rate achieved the lower packet loss ratio and vise versa.

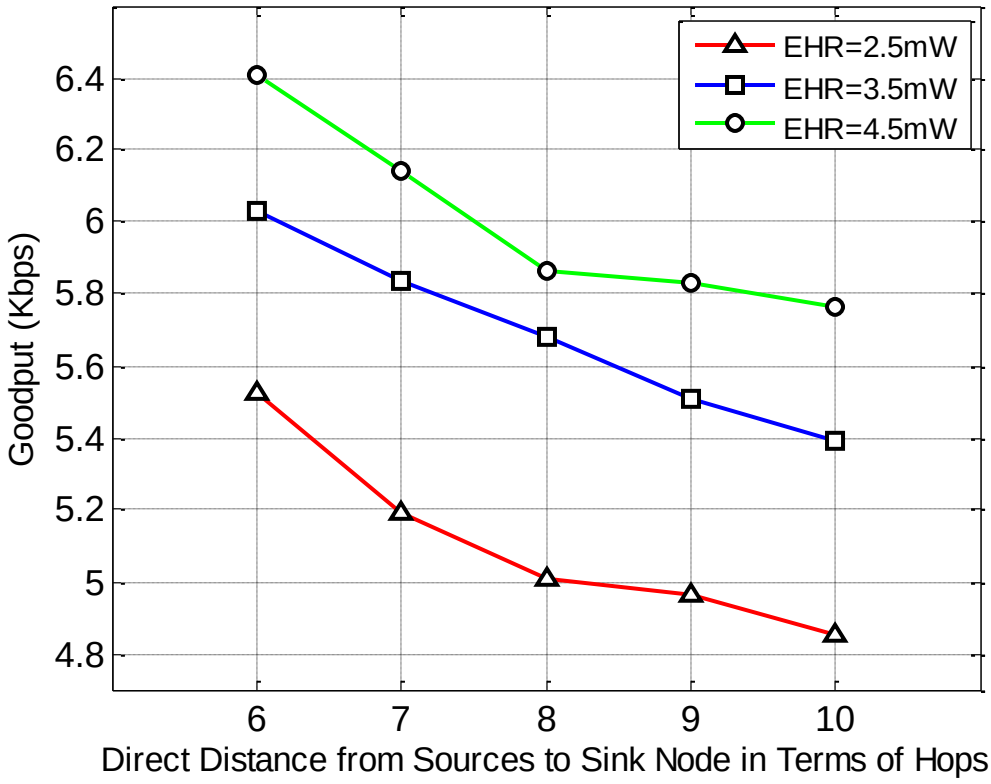


Figure 6.14 Average goodput per flow. Eighteen TCP flows at average rate of 8Kbps and file size of 209Kbytes per flow.

Figure 6.14 shows the average goodput of the TCP flows versus the number of hops. Goodput decreases as the number of hops increases. As expected, the configuration with highest energy harvesting rate achieved the highest goodput and vise versa.

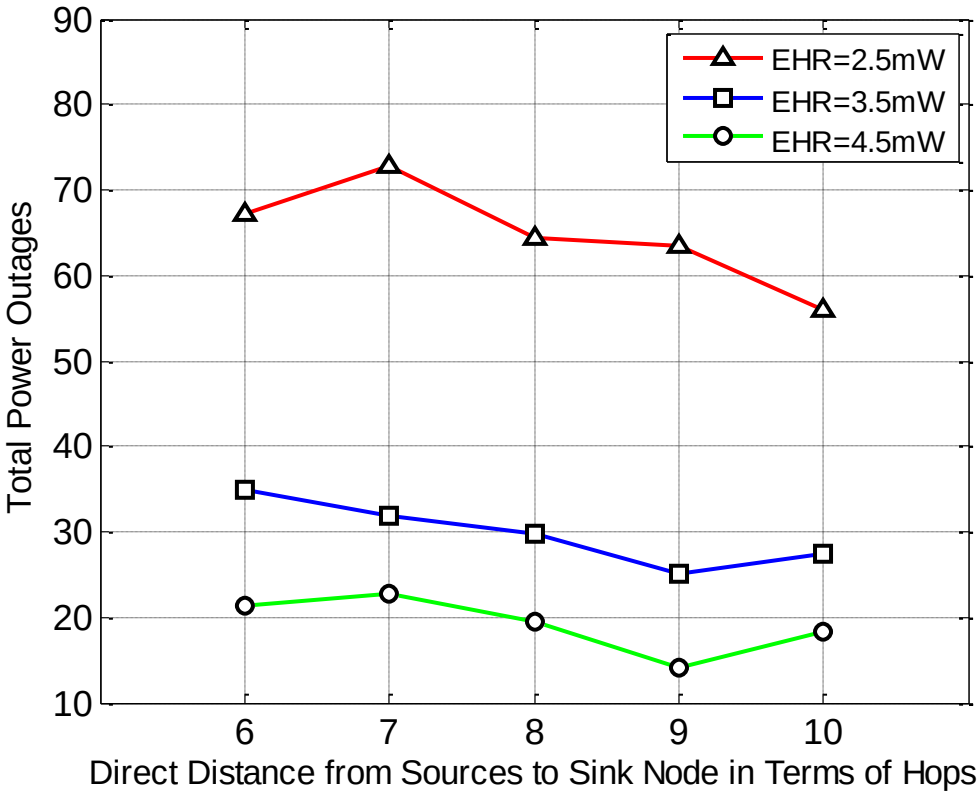


Figure 6.15 Total power outages. Eighteen TCP flows at average rate of 8Kbps and file size of 209 Kbytes per flow.

Figure 6.15 shows the total power outages in the network versus number of hops. (HER=2.5mW) achieved highest level of power outages in comparison to other configurations. This is because there are many weak nodes that cause the power outages to happen more frequently. The high level of power outages explains why the performance of the network with low harvesting rate is relatively poor in comparison to higher harvesting rate configurations.

Table 6.8 summarizes the effects of the different energy harvesting rates on the performance with reference to EHR of 2.5 mW.

Table 6.8 Computation of effects of EHR on the performance with respect to HER = 2.5 mW. (+) and (-) signs indicate increment or decrement in the corresponding performance metric, respectively.

Performance metric	EHR	
	3.5 mW (%)	4.5 mW (%)
End to end delay	-5.65	-9.94
Packet loss ratio	-8.19	-14.52
Goodput	11.41	17.51
Power outage	-54.01	-70.33

6.2.4 Studying the Effect of Varying Application Data Rate

In this section, a simulation scenario is conducted where the variable is the number of hops between the source nodes to the sink node and the application data rate. The goal of this scenario is to study the effect of the application data rate on the network performance. The routing protocol which used in this scenario is EHAODV-m. The simulation parameters are listed in Table 6.9.

Table 6.9 Simulation parameters.

Parameter	Value
Routing protocol	EHAODV-m
Application traffic	On time: 5 sec, Off time: 3 sec Rate: varying [10, 20, 40, 50] Kbps
Source to destination physical distance	Varying 6 to 10 hops

The other simulation parameters of this scenario are same as in Table 6.2.

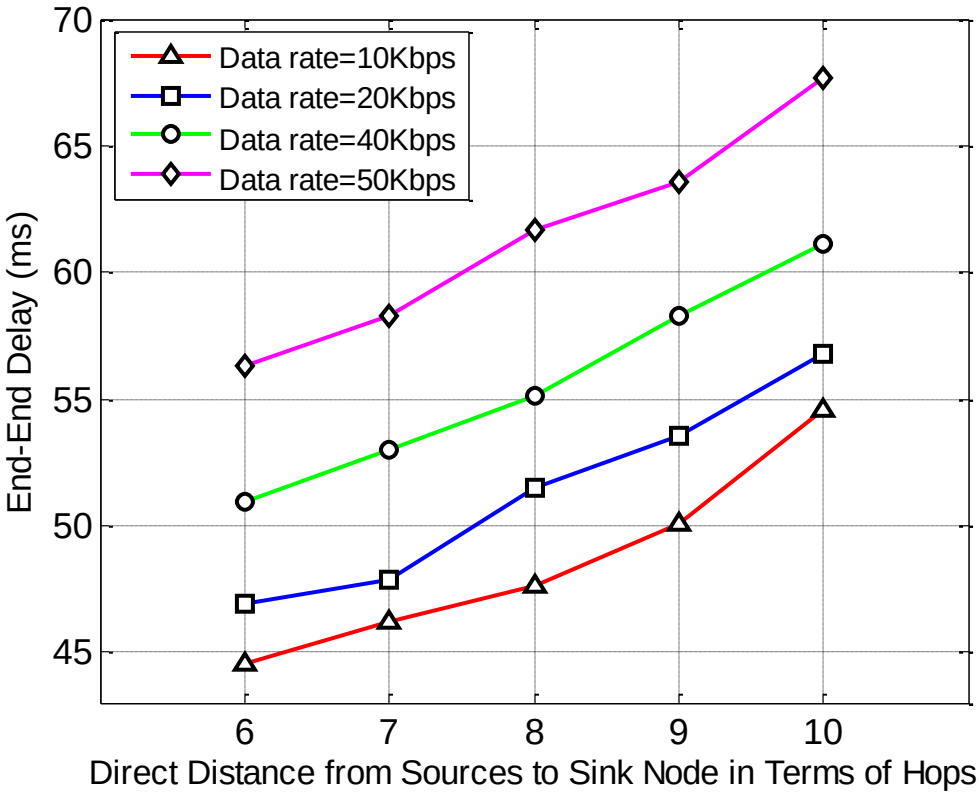


Figure 6.16 Average packet end-end delay per flow. Eighteen TCP flows with file size of 209 Kbytes per flow.

Varying the application data rate has a major effect on the delay and goodput. Figure 6.16 shows the average packet end-to-end delay of the flows versus the number of hops for different application data rates. The higher number of hops increases the propagation delay. Increasing the application data rate causes higher queuing delay. The relation between the data rate and the delay is non linear. The difference in the delay when the rate is changed from 40 Kbps to 50 Kbps is much higher than the difference in the delay when the rate is changed from 10 Kbps to 20 Kbps.

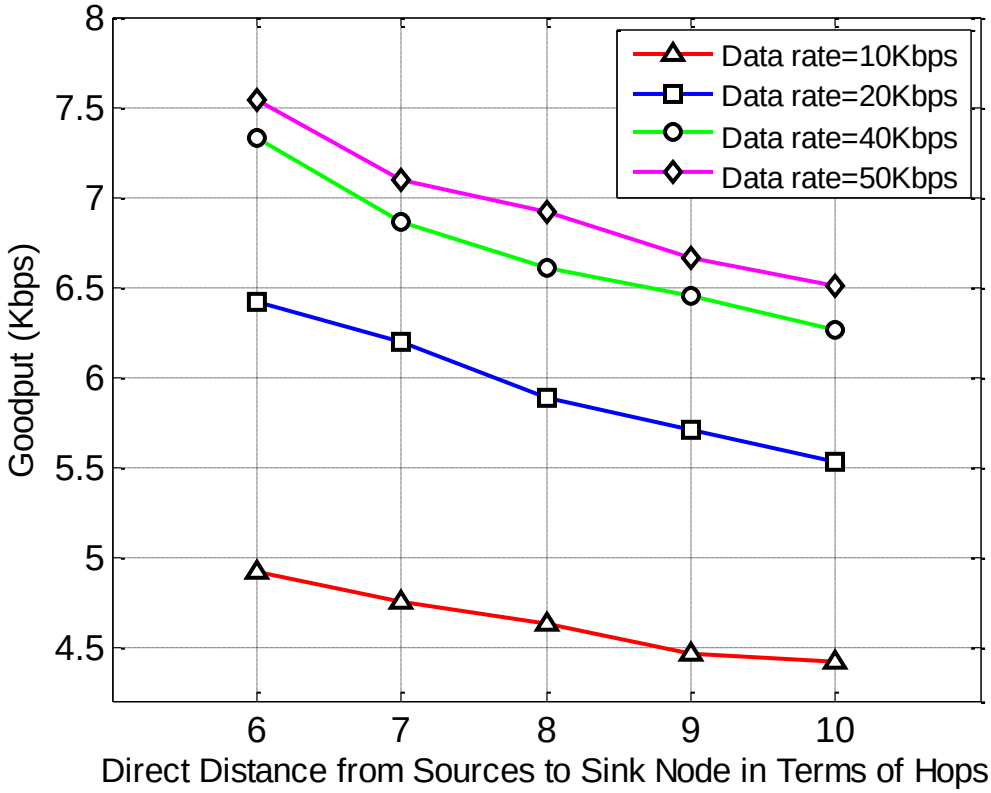


Figure 6.17 Average goodput per flow. Eighteen TCP flows with file size of 209 Kbytes per flow.

Figure 6.17 shows the average goodput of the TCP flows versus the number of hops for different application data rates. The goodput decreases as the number of hops increases. In Addition, it is expected that the goodput increases as the application data rate increases. The relation between the application data rate and the goodput is linear when the utilization of the links is low. When the utilization becomes high enough, the network can't achieve higher goodput [49].

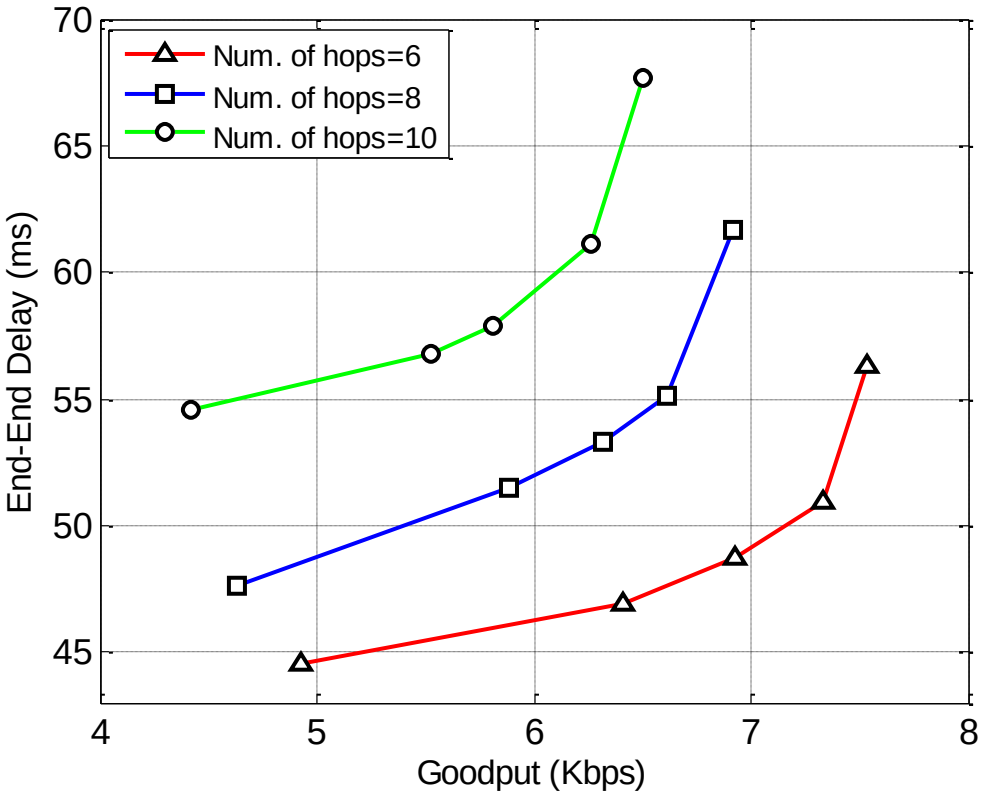


Figure 6.18 Average end-to-end delay versus goodput for different route length.

Figure 6.18 shows the delay versus the goodput for different number of hops. It is clear that the average delay increases exponentially as the network goodput approaches saturation.

If it is required to limit the delay to 55 ms, we can observe that at this delay we can achieve different levels of goodput based on the number of hops. When the number of hops is 10, the goodput is about 4.5 Kbps while it is almost doubled when the path length is 6 hops.

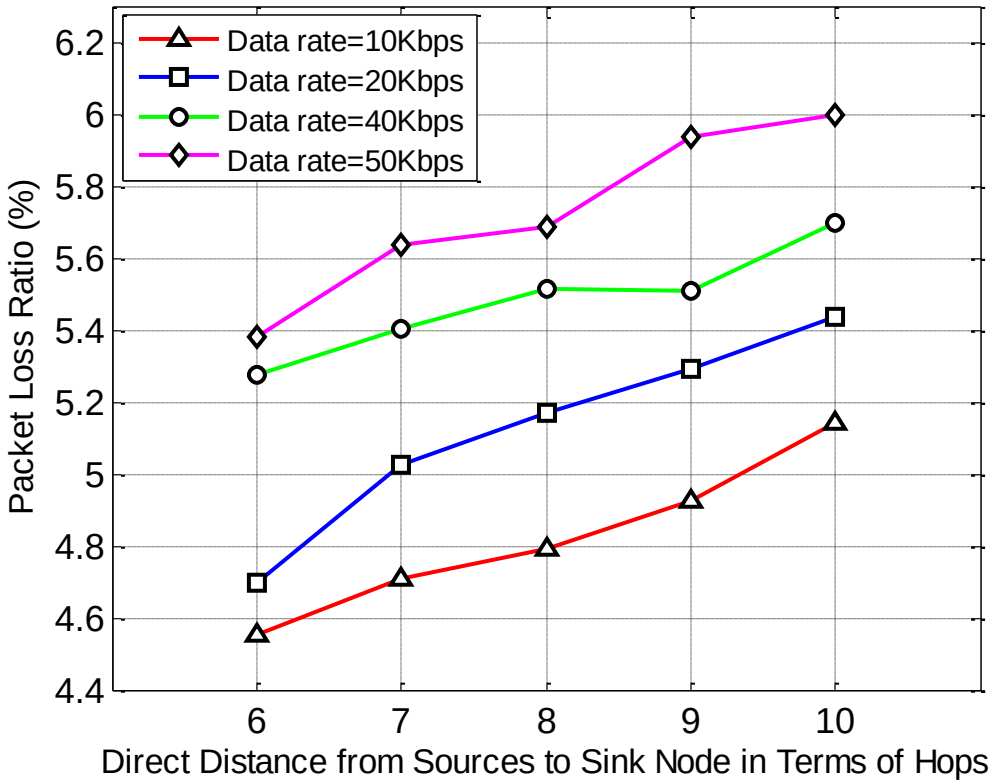


Figure 6.19 Average packet loss ratio per flow. Eighteen TCP flows with file size of 209 Kbytes per flow.

Figure 6.19 shows the average packet loss ratio of the TCP flows versus the number of hops. Packet loss ratio increases as the number of hops increases. As expected, the configuration with highest application data rate generated the highest packet loss ratio and vise versa.

Table 6.10 shows the effects of the different application data rates on the performance with reference to rate of 10 Kbps.

Table 6.10 Computation of effects of the application data rate on the performance with respect to data rate = 10 Kbps. Positive sign indicate increment in the corresponding performance metric.

Performance metric	Application data rate		
	20 Kbps (%)	40 Kbps (%)	50 Kbps (%)
End to end delay	5.59	14.57	26.58
Packet loss ratio	6.23	13.59	18.73
Goodput	28.25	44.60	49.74

6.2.5 Studying the Effect of Route Repair Mechanism

In this section, we study the full EHAODV protocol including the route repair mechanism explained before. We refer to EHAODV with route repair mechanism as EHAODV-r. The goal of the scenario is to study the performance of EHAODV-m, EHAODV-r, and original AODV. The simulation parameters are listed in Table 6.11

Table 6.11 Simulation parameters.

Parameter	Value
Routing protocol	AODV, EHAODV-m, EHAODV-r
Application traffic	Continuous TCP traffic Rate: 15 Kbps
Source to destination physical distance	Varying 6 to 10 hops
Simulation time	1000 sec
Number of packets per source	No limit

The data session in this scenario is configured to be continuous so that the route is used until a break is happen. The number of packets per source is not limited and. The other simulation parameters are the same as in Table 6.2.

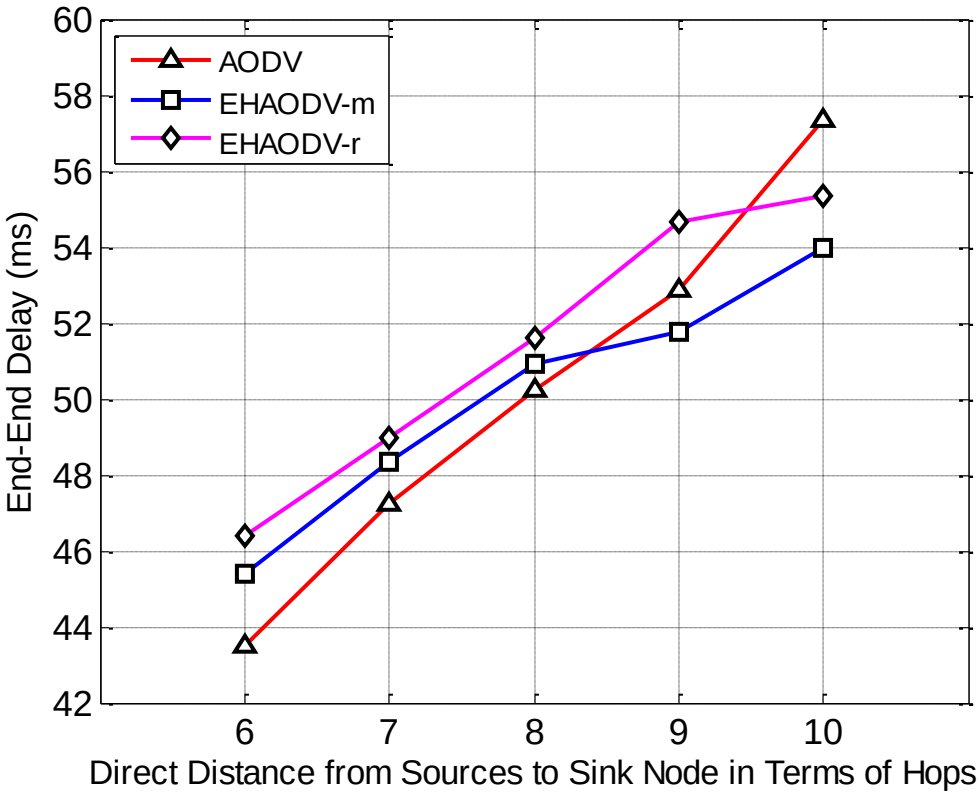


Figure 6.20 Average packet end-end delay per flow. Eighteen TCP flows at average rate of 15 Kbps and not limited file size per flow.

Figure 6.20 shows the average packet end-to-end delay versus the number of physical hops. The delay for EHAODV-r protocols achieved around 3% higher than AODV. AODV achieved lowest delay until number of hops equal eight, at number of hops equal nine, the delay of AODV where higher than EHAODV-m, and at number of hops equal ten, the delay of AODV is the highest. This is because EHAODV can find better routes when the number of hops is higher. The slight increment in EHAODV-r is caused by two reasons: the extra waiting in the source node when it receives no_delete_RERR, and the waiting time the buffered packets experience in the intermediate nodes until a new route is created.

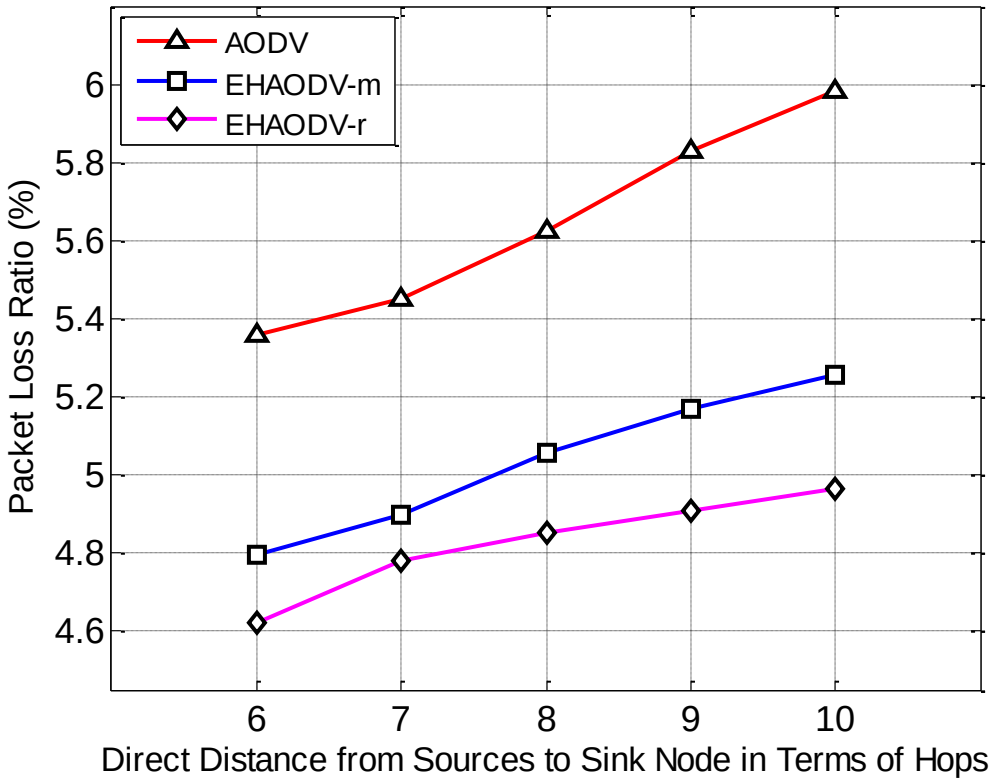


Figure 6.21 Average packet loss ratio per flow. Eighteen TCP flows at average rate of 15 Kbps and not limited file size per flow.

Figure 6.21 shows the average packet loss ratio versus the number of physical hops. EHAODV-m outperformed AODV in packet loss ratio with about 8.5 % as we saw in Figure 6.4. In this scenario, EHAODV-m is 10.9 % better than AODV. This behavior can be explained by the change in the traffic type as it is continuous while it was on/off in the other scenario. EHAODV-r is 3.7 % better than EHAODV-m. This effect is caused by the route repair mechanism as less number of packets is being lost when power outages happen.

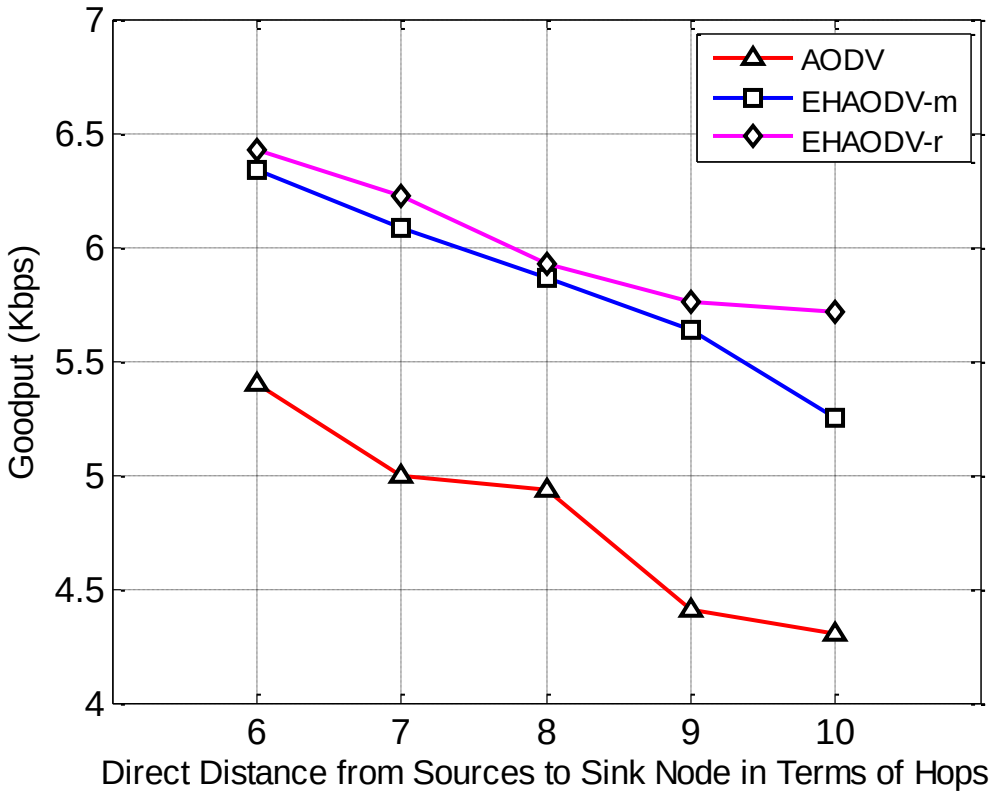


Figure 6.22 Average goodput per flow. Eighteen TCP flows at average rate of 15 Kbps and not limited file size per flow.

Figure 6.22 shows the average flow goodput versus number of hops. The goodput of EHAODV-m is about 21.4 % higher than AODV, while it shows only 12.2 % when we have on/off traffic mode as in Figure 6.5. EHAODV-r shows higher goodput than EHAODV-m with about 3 % for number of hops less than ten. But for number of hops higher than ten, it shows about 8 % improvement. The improvement can be attributed to lower number of power outages as illustrated in Figure 6.23 and resulting in lower packet loss ratio and higher goodput.

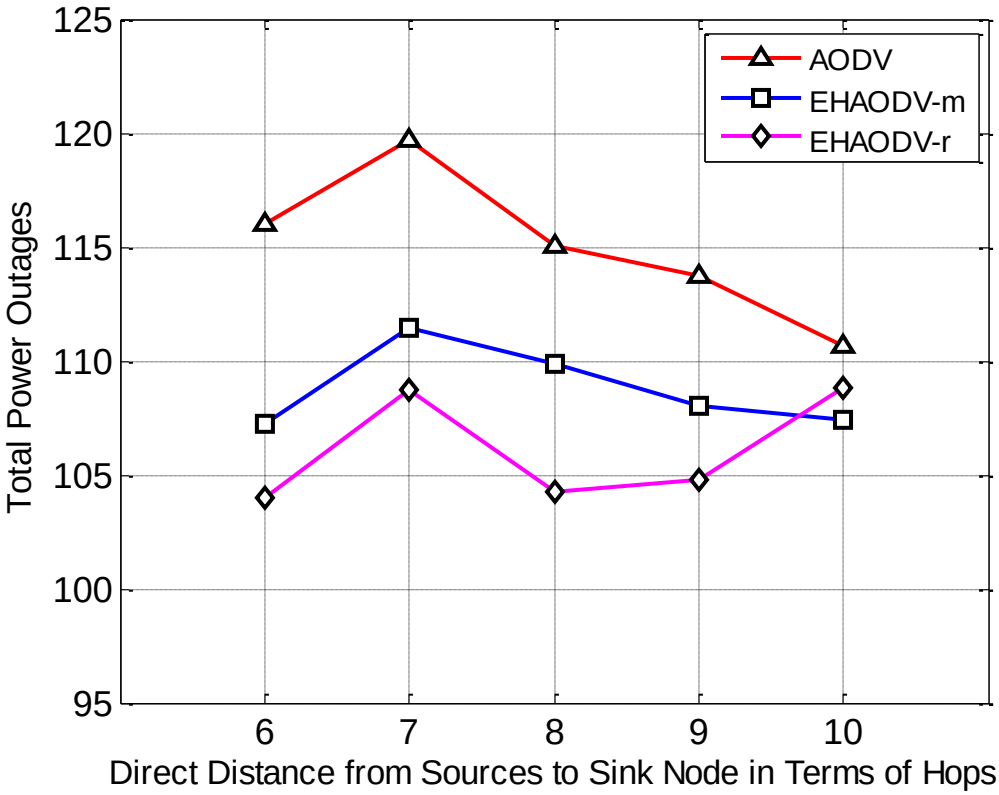


Figure 6.23 Total power outages. Eighteen TCP flows at average rate of 15 Kbps and not limited file size per flow.

Figure 6.23 shows the total power outages in the network versus number of hops. AODV achieved highest level of power outages while EHAODV-r achieved lowest level.

Table 6.12 summarizes the effects of the protocols on the different performance metrics. The percentage of enhancement or degradation with respect to AODV is shown.

Table 6.12 Computation of effects of protocols on the performance with respect to AODV. (+) and (-) signs indicate increment or decrement in the corresponding performance metric, respectively.

Performance metric	Protocol	
	EHAODV-m (%)	EHAODV-r (%)
End to end delay	-0.30	2.29
Packet loss ratio	-10.88	-14.58
Goodput	21.35	24.96
Power outage	-5.40	-7.74

6.2.6 Studying the Effect of Changing Number of Data Sources

In this scenario, a simulation experiment is held where the variable is the number of data sources. The goal of the scenario is to show the effect of number of sources on the performance of EHAODV-m in terms of end-to-end delay, goodput, number of power outages, and packet loss ratio. Changing the number of data sources affects the traffic load in the network, and hence the total consumed energy. The simulation parameters are listed in Table 6.13. The data session in this scenario is configured to be continuous so that the route is used until a break happens. The number of packets per source is not limited; other simulation parameters are the same as in Table 6.2.

Table 6.13 Simulation parameters.

Parameter	**Value**
Routing protocol	EHAODV-m
Application traffic	Continuous TCP traffic Rate: 10 Kbps
Source to destination physical distance	10 hops
Number of packets per source	No limit
Number of data sources	Varying: 3, 6, 9, 12, 15

Figure 6.24 shows the average packet end-to-end delay of the TCP flows versus number of data sources. The end-to-end delay proportionally increases with the number of data sources. Having more data sources in the network causes higher energy consumption, and hence, more power outages. The number of physical hops from the source to sink node is fixed in this scenario. In designing an energy-harvesting network where the energy is a substantial resource, the traffic load should be taken in account with regard to the available energy.

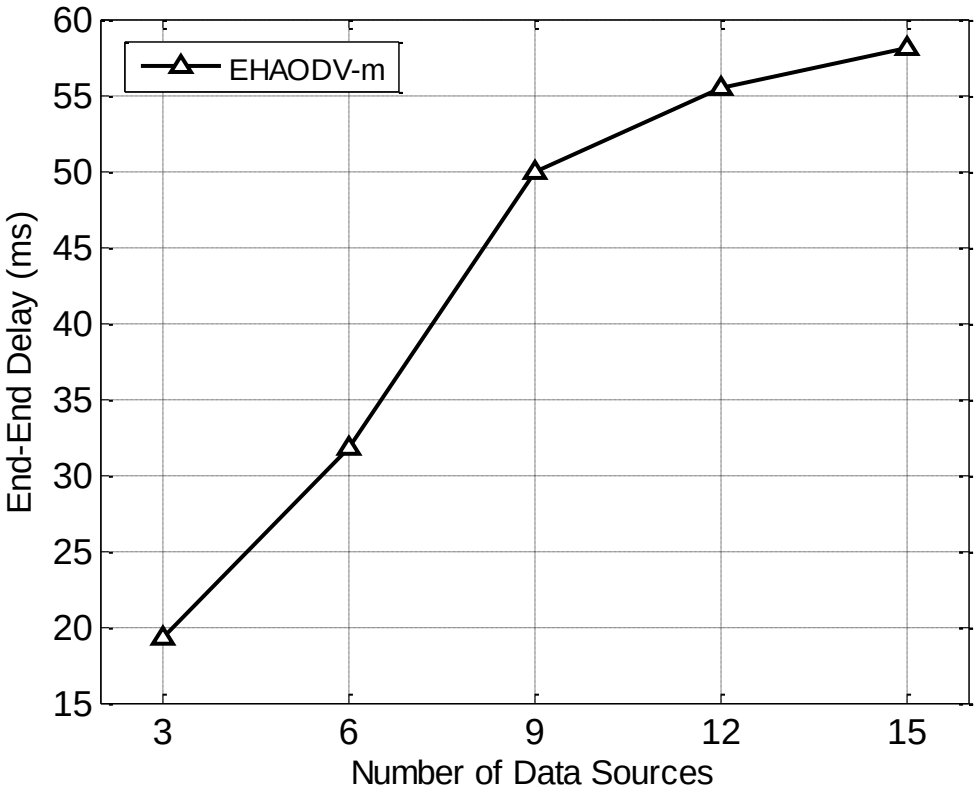


Figure 6.24 Average packet end-to-end delay per flow. Variable number of TCP flows at average rate of 10 Kbps and a non limited file size per flow and number of hops of 10.

Figure 6.25 shows the average packet loss ratio of the TCP flows versus the number of data sources. Packet loss ratio increases as the number of data sources increases. When the network experience higher number of power outages, this causes a higher number of packets to be lost.

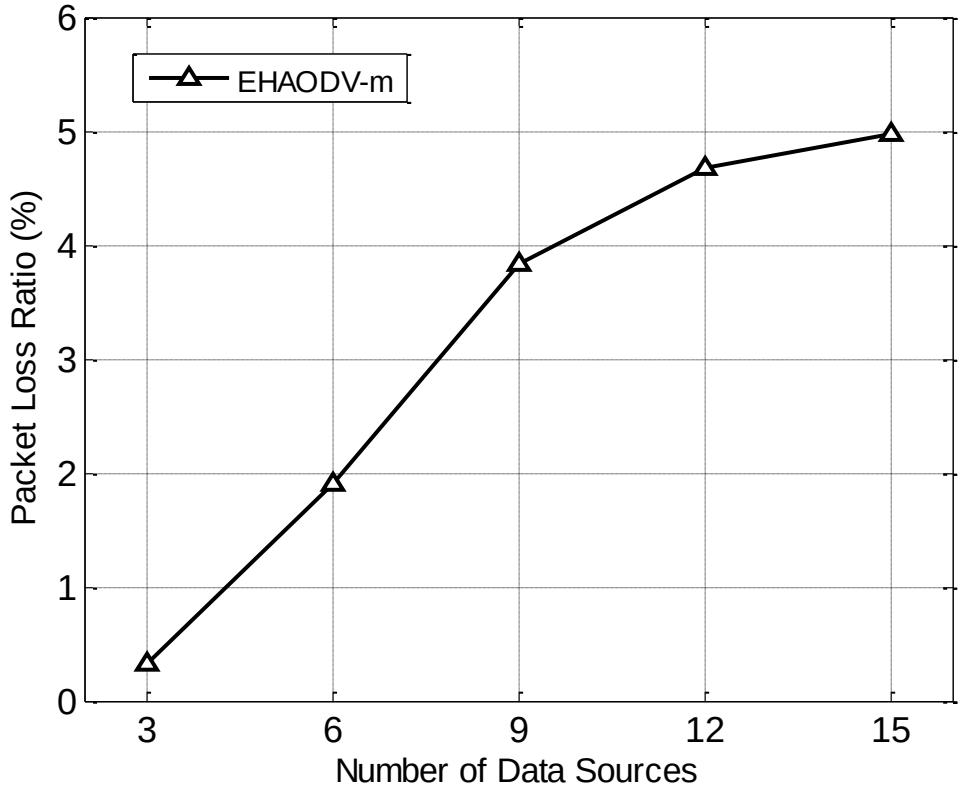


Figure 6.25 Average packet loss ratio per flow. Variable number of TCP flows at average rate of 10 Kbps and a non limited file size per flow and number of hops of 10.

Figure 6.26 shows the number of power outages in the network versus the number of data sources. As expected, a higher traffic load in the network causes more nodes to die. The power outages degraded the network performance as shown.

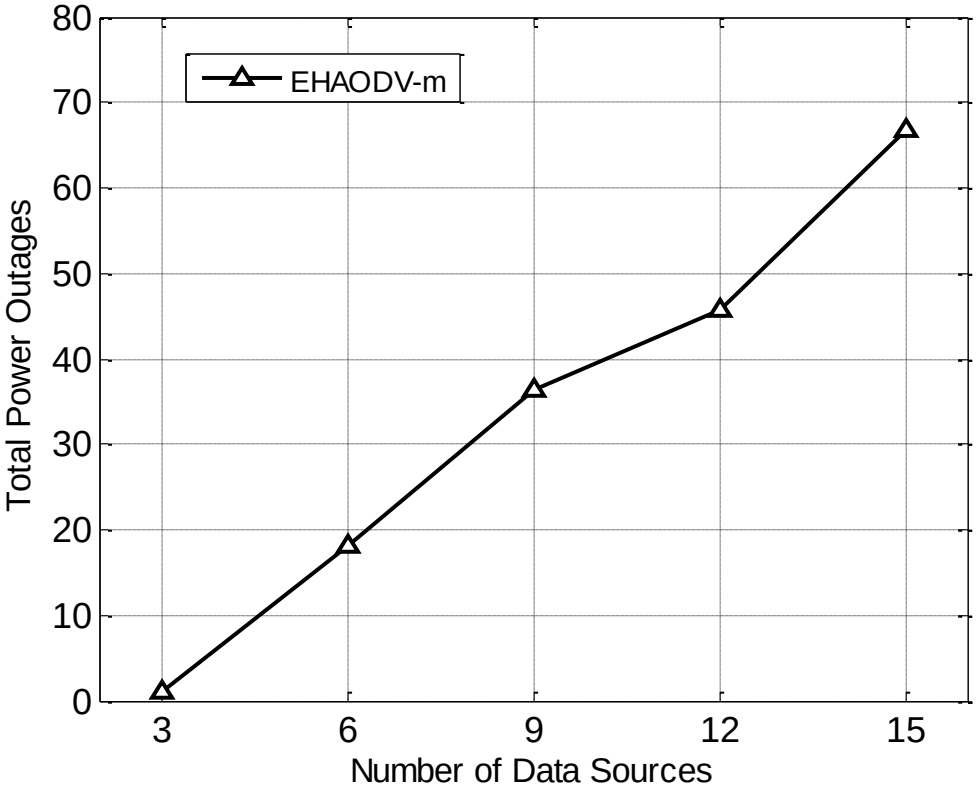


Figure 6.26 Number of power outages in the network. Variable number of TCP flows at average rate of 10 Kbps and a non limited file size per flow and number of hops of 10.

Figure 6.27 shows the average goodput of the TCP flows versus the number of data sources. The goodput is decreasing as the number of data sources is increasing. Each time a power outage occurs in the network, it interrupts the data flow that goes through the died node. It also causes the source node to wait until it realizes the error in the route and then start a new route discovery process. The goodput decreased significantly because of that.

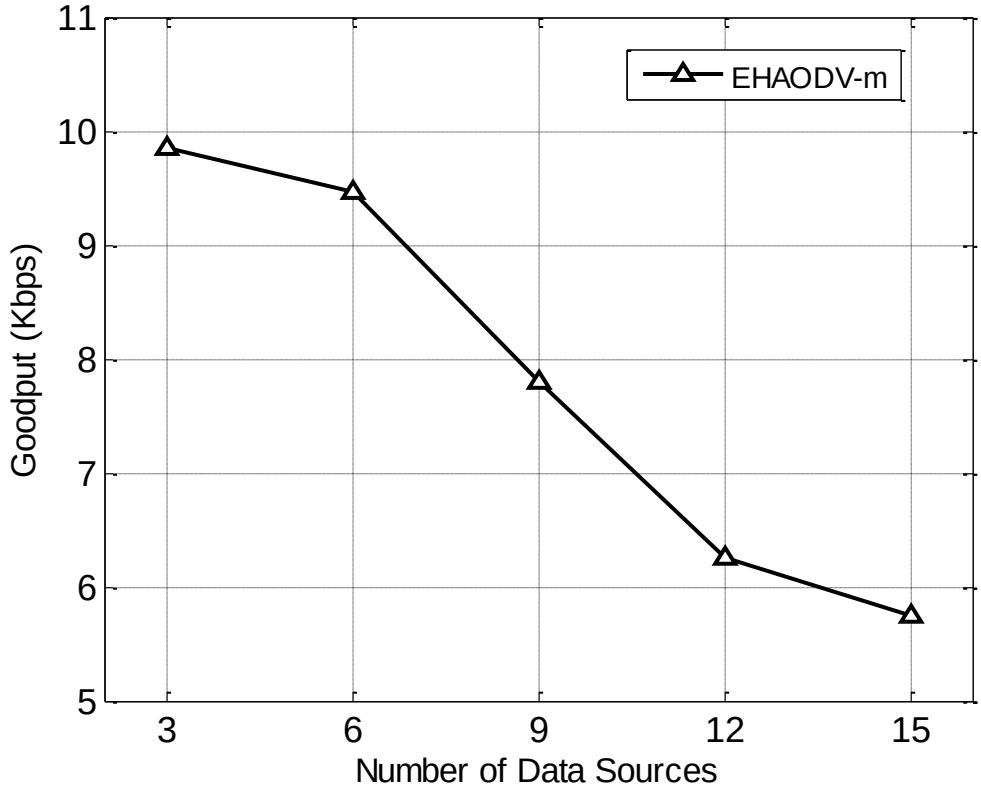


Figure 6.27 Average goodput per flow. Variable number of TCP flows at average rate of 10 Kbps and a non limited file size per flow.

In order to understand the behavior of the node during the simulation and see how the power outage affects the goodput, one of the eighteen TCP flows was traced versus time. Figure 6.28 show the power outages occurrences in the network for a single run and also shows the active data dissemination intervals of same node. The node started sending data to the sink around time 12s and continued sending until time 186s. The first power outage in the network occurred at time 120s; it had no effect on the traced flow. What caused to flow to stop is the fifth outage that occurred at time 186s. At time 240s, the node resumed sending data after route discovery process until time 257s when the flow was interrupted again because of another power outage in the network. This behavior

continues until the end of simulation. We notice that each time the flow is interrupted; it was interrupted because of a power outage. On the other hand, not all of the power outages cause the flow to be interrupted since it might happen in nodes that are not participating in the flow.

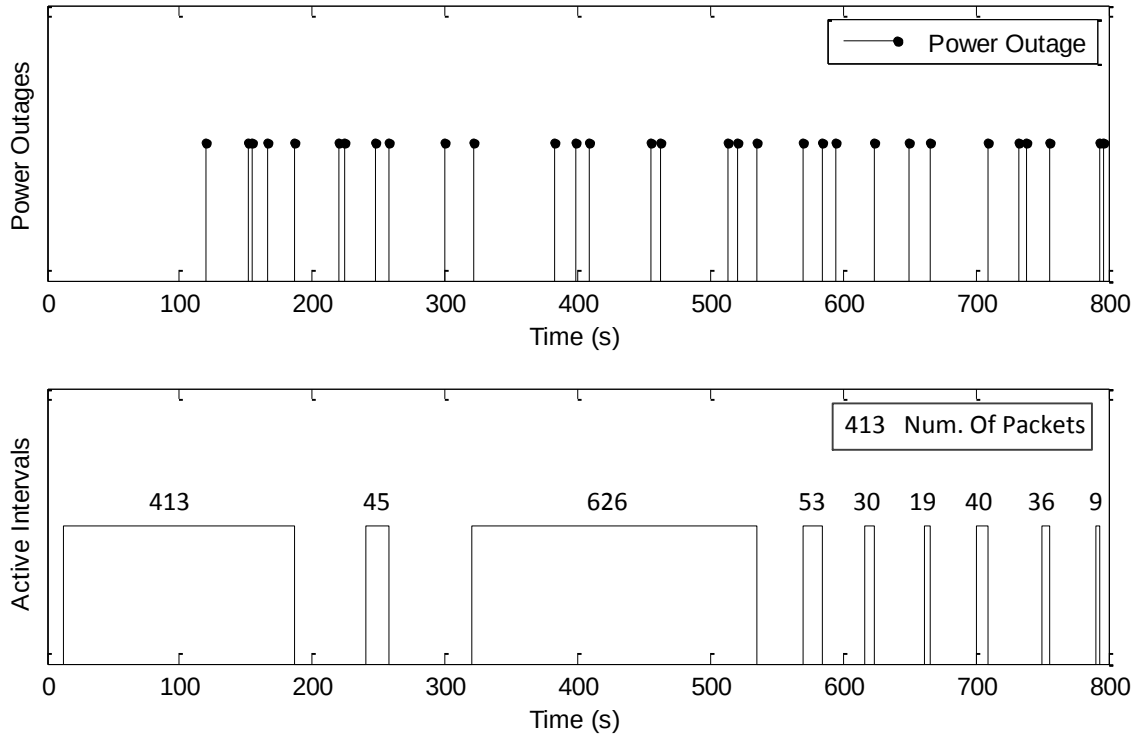


Figure 6.28 Power outages in the network versus time and active data dissemination intervals of a single node.

The number above each data dissemination interval represents the total number of packets that are received during that interval. The total number of received packets in all intervals is 1271 packets. Multiplying number of packets by packet size gives the total bytes which is 665.3 Kbytes. The flow time in ns-3 is measured by the time the first packet is received until the time the last packet is received which equals in this scenario

781 seconds. Therefore, the goodput of this scenario is 6.98 Kbps where the data generation rate is 10 Kbps.

The sum of active time intervals equals 449s. Then the goodput for the active intervals only equals 12.14 Kbps. The node generates data at rate of 10 Kbps, when there is no route to sink node, the node buffers the generated data. Once a route is created, the node sends all its buffered packets. This caused the goodput of the active time intervals increases than the actual data generating rate.

The fact that the data flow is being interrupted causes the goodput to be less than the configured sending data rate. The number of bytes (N) that is assumed to take T seconds based on the sending data rate is taking more than T seconds because of the interrupts in the flow. The goodput is calculated by dividing the total received bytes over the total flow time. Having periods of idle intervals in the flow causes a drop in the goodput as it is shown in all scenarios. Having higher number of sources, higher data rate or less energy in the network result in higher number of power outages and hence higher drop in the goodput and the performance in general.

6.2.7 Studying the Effect of Network Topology on the Performance

In this scenario, we consider random nodes deployment. The goal of this scenario is to study the effect of the random topology on EHAODV performance. In all previous scenarios, a grid topology is assumed as shown in Figure 6.1. Figure 6.29 shows a snapshot of a random topology. In order to have a fair comparison with grid topology, we have fixed the positions of both the source nodes and the sink nodes as in the grid topology. The scenario has been studied using six different random topologies and each topology is repeated 30 times in order to achieve 95% confidence level.

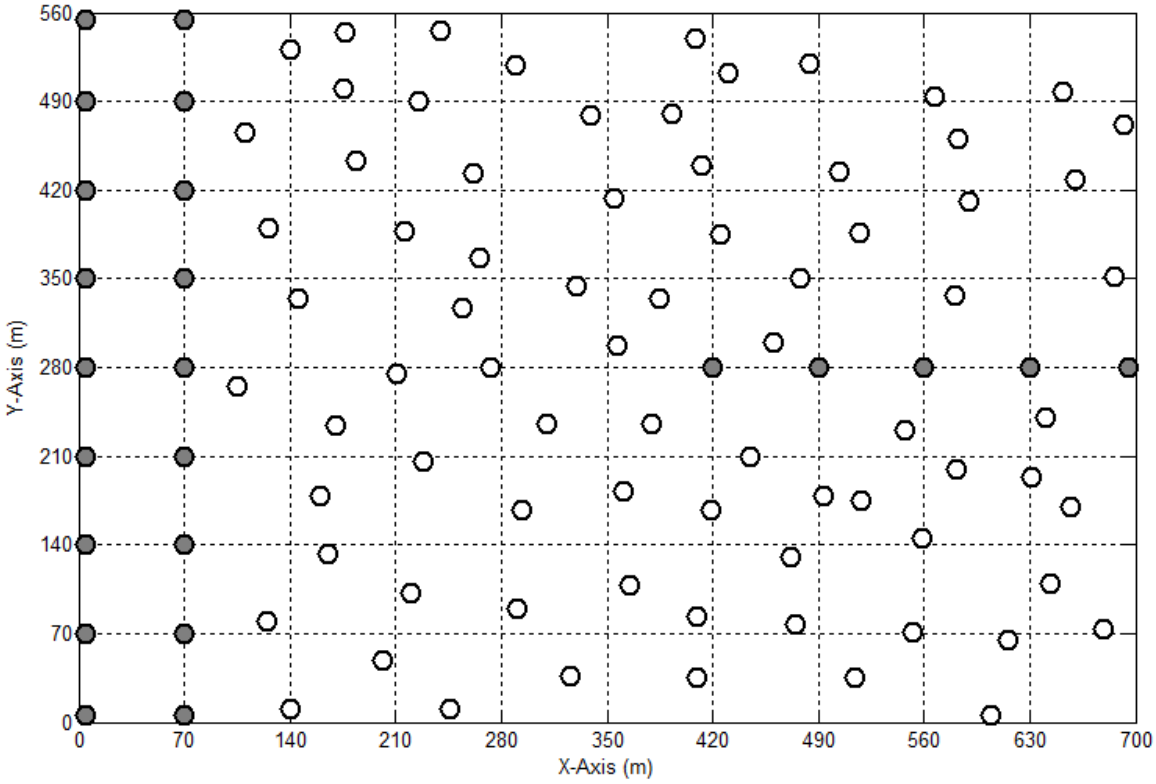


Figure 6.29 Random network topology.

The node transmission range is 100 meters. The network is designed to be connected (i.e. no isolated nodes). One sink application is installed at a time on one of the gray nodes on the right depending on the targeted physical distance between the source

node and destination node. Based on the number of physical hops from the source to destination, the sink application is installed either on the closest or the furthest node. The first eighteen nodes (gray) on the left are the source nodes. The source and sink nodes locations are fixed similar to grid topology to guarantee the same physical distance between the sources and sink nodes.

Table 6.14 Simulation parameters.

Parameter	**Value**
Routing protocol	EHAODV-m
Application traffic	On time: 5 sec, Off time: 3.1 sec Rate: 13 Kbps, Transport: TCP
Source to destination physical distance	Varying 6 to 10 hops
Number of packets per source	400 packets
Number of data sources	18
Number of topologies (replications)	6

The simulation parameters are listed in Table 6.14. This scenario is similar to scenario 6.2.1 except the random topology. The other simulation parameters are the same as in Table 6.2. The scenario was run on six different random topologies.

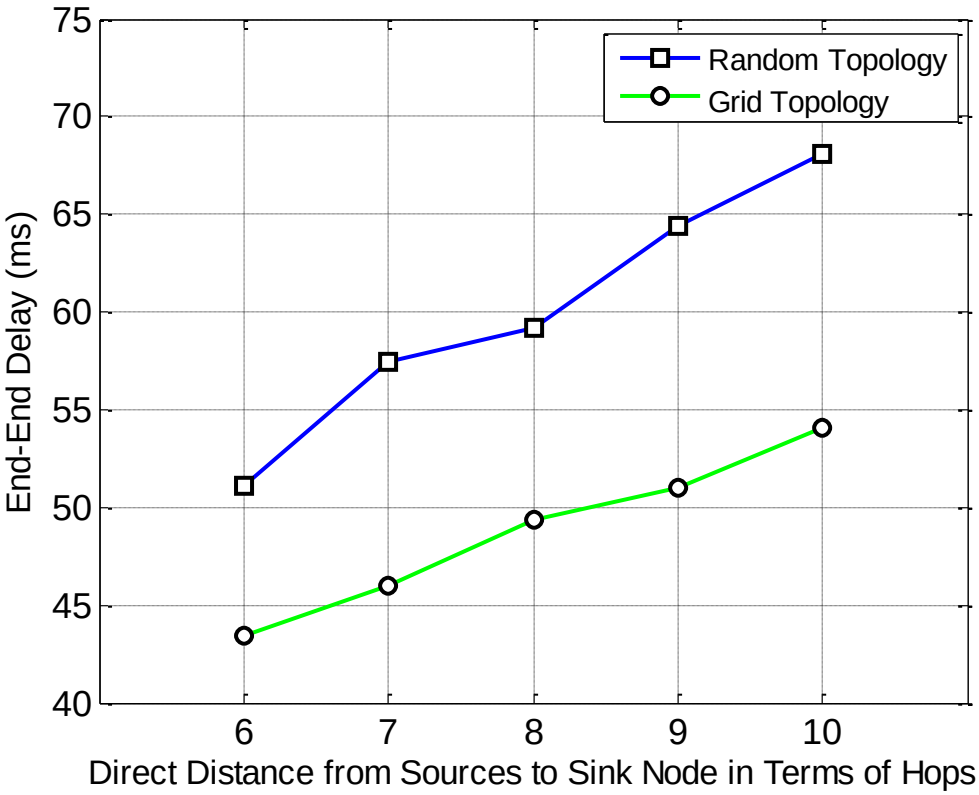


Figure 6.30 Average packet end-end delay per flow. Eighteen TCP flows at average rate of 8 Kbps and file size of 209 Kbytes per flow with random network topology.

Figure 6.30 shows the average packet end-to-end delay versus the number of physical hops. It is expected that the value of end-to-end delay increases when the number of hops is increased. The end-to-end delay of the network with random topology is higher than it is with a grid topology by about 23%. Since the network connection is random, the path from the source to destination might take higher number of hops.

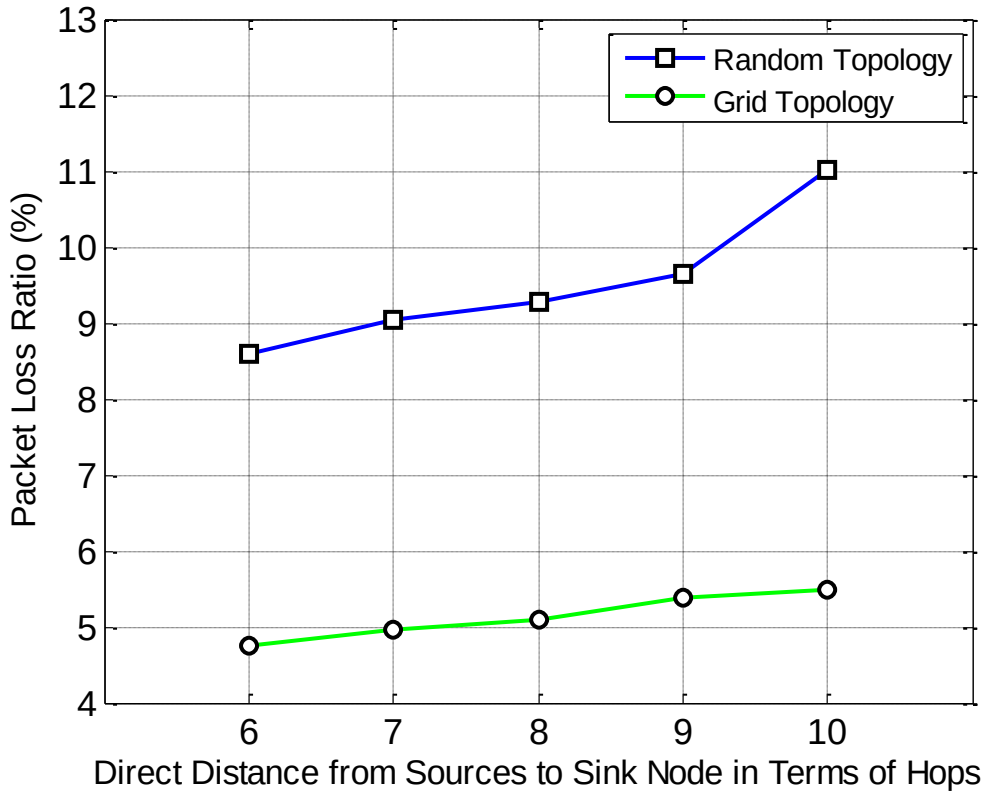


Figure 6.31 Average packet loss ratio per flow. Eighteen TCP flows at average rate of 8Kbps and file size of 209 Kbytes per flow with random network topology.

Figure 6.31shows the average packet loss ratio versus the number of physical hops. The packet loss ratio in the random topology is around 85% higher than it is in the grid topology. In addition, the loss ratio in random topology is increasing in higher rate than it is in the grid topology.

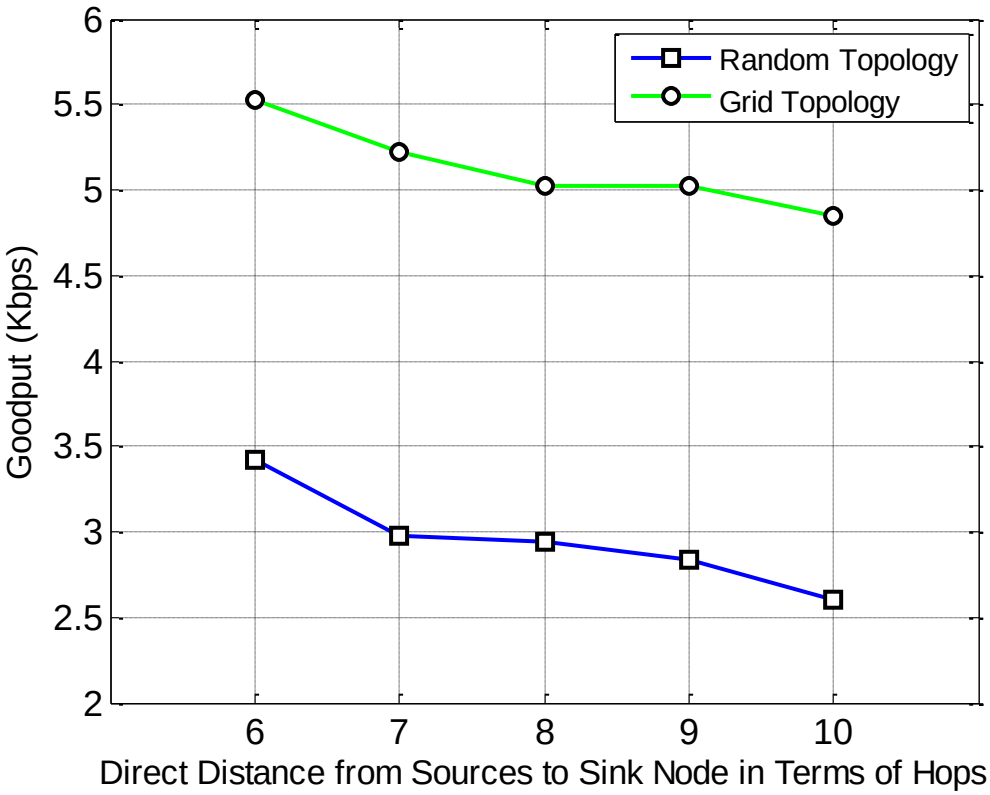


Figure 6.32 Average goodput per flow. Eighteen TCP flows at average rate of 8Kbps and file size of 209 Kbytes per flow with random network topology.

Figure 6.32 shows the average flow goodput versus number of hops. The goodput decreases when the hop-count increases. The random network topology achieved lower goodput than the grid topology.

Table 6.15 summarizes the effects of the network topology on the different performance metrics. The percentage of the enhancement or degradation with respect to grid topology is shown.

Table 6.15 Computation of effects of network topology on the performance with respect to grid topology. (+) and (-) signs indicate increment or decrement in the corresponding performance metric, respectively.

Performance metric	Random network Topology (%)
End to end delay	23.1
Packet loss ratio	85.4
Goodput	-42.4

CHAPTER 7

CONCLUSION AND FUTURE WORK

This chapter summarizes the thesis work and its contributions. In our research, we have tackled important problems in energy-harvesting based WSNs that are, first, choosing the best route based on general metrics including node's remaining energy and energy harvesting rate in order to achieve highest network availability. Second, we proposed a repairing mechanism for broken routes in order to save the ongoing route and save packets in intermediate nodes from being lost.

7.1 Conclusion

In this thesis, we proposed two algorithms to choose the best path based on current node's energy, energy harvesting rate, and path length. The first algorithm is EHAODV-a, in which the node average energy across the path has first priority in choosing the best routes. This algorithm increased the flow goodput about 16% and decreased the flow packet loss ratio about 13.5% in reference to original AODV, while it caused an increment in the flow delay about 8.6%.

The second algorithm is EHAODV-m, in which we build the path such that we avoid the potential bottleneck nodes (i.e. nodes that has minimum energy). Therefore, this approach selects the path that has the maximum-minimum energy nodes. The simulation results demonstrated that this algorithm have enhanced the flow goodput by about 12.2% and have decreased the flow packet loss ratio about 8.5% in reference to original AODV, while the flow delay have increased by about 2.9%.

In addition, we proposed a local route repair mechanism to reduce the packet loss ratio in the network. in this mechanism, the node upstream of a broken route don't invalidate the route directly after receiving a RERR; instead, the node upstream starts a route discovery to the destination and resume forwarding the buffered packets while the source node starts using the newly constructed route. This mechanism have decreased the flow packet loss ratio about 3.7%, increased the flow goodput about 3.6%, and decreased the total power outages in the network by about 2.3% in comparison to EHAODV-m. On the other hand, it has caused the flow delay to increase about 2.6%.

7.2 Future Work

In EHAODV, we proposed the node's remaining energy, node's energy harvesting rate, and node's average energy as metrics for route selection. We suggest adding a new metric that represents the node activity history such as number of power outages.

EHAODV is a reactive routing protocol i.e. creates the route whenever needed only. The implementation of same metrics of EHAODV on a proactive protocol will give different result so it is worth to be tested.

EHAODV is a single path routing protocol i.e. only one route is built from the source to destination. AOMDV is a multipath version of AODV protocol. Having many routes to the same destination can reduce the route discovery delay overhead but adds a route maintenance overhead.

The route selection in EHAODV is implemented by normal programming i.e. conditional logic programming where variables may take on true or false values. Using of fuzzy logic in route selection will ease the route selection and make it more intelligent.

Fuzzy logic is a form of many-valued logic which deals with reasoning that is approximate rather than fixed and exact.

The route repair mechanism that proposed in this work is reactive i.e. the route break take place then the route is repaired. A proactive route repair mechanism can exploit the energy information of the nodes so it discovers a new route before a break happen.

Finally, the data requests of the nodes in this work are exhaustive. The nodes continue sending data to the sink even when the network is not capable to serve those requests because of limited energy resource. A scheduling algorithm can avoid this problem by allowing the nodes to send data only when the network is capable to serve those requests.

References

[1] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, "Next century challenges: Scalable coordination in sensor networks," in *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, 1999, pp. 263-270.

[2] P. Corke, T. Wark, R. Jurdak, W. Hu, P. Valencia, and D. Moore, "Environmental wireless sensor networks," *Proceedings of the IEEE,* vol. 98, pp. 1903-1917, 2010.

[3] R. J. Vullers, R. Schaijk, H. J. Visser, J. Penders, and C. V. Hoof, "Energy harvesting for autonomous wireless sensor networks," *Solid-State Circuits Magazine, IEEE,* vol. 2, pp. 29-38, 2010.

[4] M. Winkler, M. Street, K.-D. Tuchs, and K. Wrona, "Wireless sensor networks for military purposes," in *Autonomous Sensor Networks*, ed: Springer, 2013, pp. 365-394.

[5] Wikipedia, "Wireless Sensor Network. Available: http://en.wikipedia.org/wiki/Wireless_sensor_network," ed, 2015.

[6] I. Cisco Systems, *Internetworking Technologies Handbook*: Cisco Press, 2004.

[7] J. Li, C. Blake, D. S. De Couto, H. I. Lee, and R. Morris, "Capacity of ad hoc wireless networks," in *Proceedings of the 7th annual international conference on Mobile computing and networking*, 2001, pp. 61-69.

[8] C. de Morais Cordeiro and D. P. Agrawal, "Mobile ad hoc networking," *Center for Distributed and Mobile Computing, ECECS, University of Cincinnati,* 2002.

[9] C. Mbarushimana and A. Shahrabi, "Comparative study of reactive and proactive routing protocols performance in mobile ad hoc networks," in *Advanced Information Networking and Applications Workshops, 2007, AINAW'07. 21st International Conference on*, 2007, pp. 679-684.

[10] C. Perkins, E. Belding-Royer, and S. Das, *Ad hoc On-Demand Distance Vector (AODV) Routing*: RFC Editor, 2003.

[11] W. K. Seah, Z. A. Eu, and H.-P. Tan, "Wireless sensor networks powered by ambient energy harvesting (WSN-HEAP)-Survey and challenges," in *Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology, 2009. Wireless VITAE 2009. 1st International Conference on*, 2009, pp. 1-5.

[12] X. Fang, S. Misra, G. Xue, and D. Yang, "Smart grid—The new and improved power grid: A survey," *Communications Surveys & Tutorials, IEEE,* vol. 14, pp. 944-980, 2012.

[13] S. Jabbar, A. E. Butt, and A. Minhas, "Threshold based load balancing protocol for energy efficient routing in WSN," in *Advanced Communication Technology (ICACT), 2011 13th International Conference on*, 2011, pp. 196-201.

[14] X. Li, W. Gang, L. Zongqi, and Z. Yanyan, "An energy-efficient routing protocol based on particle swarm clustering algorithm and inter-cluster routing algorithm for WSN," in *Control and Decision Conference (CCDC), 2013 25th Chinese*, 2013, pp. 4029-4033.

[15] L. Mengyao, Y. Zhang, and X. Li, "Ring-based security energy-efficient routing protocol for WSN," in *Control and Decision Conference (2014 CCDC), The 26th Chinese*, 2014, pp. 1892-1897.

[16] L. Ya, W. Pengjun, L. Rong, Y. Huazhong, and L. Wei, "Reliable energy-aware routing protocol for heterogeneous WSN based on beaconing," in *Advanced Communication Technology (ICACT), 2014 16th International Conference on*, 2014, pp. 109-112.

[17] W. K. Seah, Y. Tan, and A. T. Chan, *Research in energy harvesting wireless sensor networks and the challenges ahead*: Springer, 2013.

[18] M. A. Green, K. Emery, Y. Hishikawa, W. Warta, and E. D. Dunlop, "Solar cell efficiency tables (version 39)," *Progress in photovoltaics: research and applications,* vol. 20, pp. 12-20, 2012.

[19] I. Mathews, G. Kelly, P. J. King, and R. Frizzell, "GaAs solar cells for Indoor Light Harvesting," in *Photovoltaic Specialist Conference (PVSC), 2014 IEEE 40th*, 2014, pp. 0510-0513.

[20] A. Cammarano, C. Petrioli, and D. Spenza, "Pro-Energy: A novel energy prediction model for solar and wind energy-harvesting wireless sensor networks," in *Mobile Adhoc and Sensor Systems (MASS), 2012 IEEE 9th International Conference on*, 2012, pp. 75-83.

[21] H. Kim, Y.-J. Min, C.-H. Jeong, K.-Y. Kim, C. Kim, and S.-W. Kim, "A 1-mW Solar-Energy-Harvesting Circuit Using an Adaptive MPPT With a SAR and a Counter," *Circuits and Systems II: Express Briefs, IEEE Transactions on,* vol. 60, pp. 331-335, 2013.

[22] H. Sherif, M. Eltaib, and A. Alsuwaiyan, "On the Design of High Power Low Frequency Harvesters for Car Engine," SAE Technical Paper2013.

[23] B. Jiang, K. Cao, L. Chen, H. Chen, H. Zhang, and Q. Wang, "Low-power design of a self-powered piezoelectric energy harvesting system," in *Control Conference (CCC), 2014 33rd Chinese*, 2014, pp. 6937-6940.

[24] R. Meier, N. Kelly, O. Almog, and P. Chiang, "A piezoelectric energy-harvesting shoe system for podiatric sensing," in *Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE*, 2014, pp. 622-625.

[25] Y. K. Tan and S. K. Panda, "Energy harvesting from hybrid indoor ambient light and thermal energy sources for enhanced performance of wireless sensor nodes," *Industrial Electronics, IEEE Transactions on,* vol. 58, pp. 4424-4435, 2011.

[26] V. Leonov, "Thermoelectric energy harvesting of human body heat for wearable sensors," *Sensors Journal, IEEE,* vol. 13, pp. 2284-2291, 2013.

[27] H. Wu, S. Nabar, and R. Poovendran, "An energy framework for the network simulator 3 (NS-3)," in *Proceedings of the 4th International ICST Conference on Simulation Tools and Techniques*, 2011, pp. 222-230.

[28] R. Doost, K. R. Chowdhury, and M. Di Felice, "Routing and link layer protocol design for sensor networks with wireless energy transfer," in *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, 2010, pp. 1-5.

[29] C.-W. Tan and S. K. Bose, "Modifying AODV for efficient power-aware routing in MANETs," in *TENCON 2005 2005 IEEE Region 10*, 2005, pp. 1-6.

[30] M. Lotfi, S. Jabbehdari, and M. A. Shahmirzadi, "A new energy efficient routing algorithm based on a new cost function in wireless ad hoc networks," *arXiv preprint arXiv:1006.4557,* 2010.

[31] T. Poongkuzhali, V. Bharathi, and P. Vijayakumar, "An optimized power reactive routing based on AODV protocol for Mobile Ad-hoc network," in *Recent Trends in Information Technology (ICRTIT), 2011 International Conference on*, 2011, pp. 194-199.

[32] A. M. Alshanyour and U. Baroudi, "Bypass AODV: improving performance of ad hoc on-demand distance vector (AODV) routing protocol in wireless ad hoc networks," in *Proceedings of the 1st international conference on Ambient media and systems*, 2008, p. 17.

[33] W. Li, M. Chen, and M.-m. Li, "An enhanced aodv route protocol applying in the wireless sensor networks," in *Fuzzy Information and Engineering Volume 2*, ed: Springer, 2009, pp. 1591-1600.

[34] P. Gong, Q. Xu, and T. M. Chen, "Energy Harvesting Aware routing protocol for wireless sensor networks," in *Communication Systems, Networks & Digital Signal Processing (CSNDSP), 2014 9th International Symposium on*, 2014, pp. 171-176.

[35] S. Tennina, M. Tiloca, J.-H. Hauer, M. Bouroche, M. Alves, A. Koubaa*, et al.*, "Snapshot of the IEEE 802.15. 4 and ZigBee Protocols," in *IEEE 802.15. 4 and ZigBee as Enabling Technologies for Low-Power Wireless Systems with Quality-of-Service Constraints*, ed: Springer, 2013, pp. 3-25.

[36] (2015). *Network Simulator 3*. Available: http://www.nsnam.org/

[37] "ns-3 project, Available: https://www.nsnam.org/docs/models/html/flow-monitor.html, 2015."

[38] P. Oppenheimer, "Top-Down Network Design. A system analysis approach network design," ed: Cisco Press, 2010.

[39] N. Saputro, K. Akkaya, and S. Uludag, "A survey of routing protocols for smart grid communications," *Computer Networks,* vol. 56, pp. 2742-2771, 2012.

[40] C. E. Perkins, *Ad hoc networking*: Addison-Wesley Professional, 2008.

[41] S. Ehsan and B. Hamdaoui, "A survey on energy-efficient routing techniques with QoS assurances for wireless multimedia sensor networks," *Communications Surveys & Tutorials, IEEE,* vol. 14, pp. 265-278, 2012.

[42] M. K. Marina and S. R. Das, "Ad hoc on-demand multipath distance vector routing," *ACM SIGMOBILE Mobile Computing and Communications Review,* vol. 6, pp. 92-93, 2002.

[43] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA'99. Second IEEE Workshop on*, 1999, pp. 90-100.

[44] "(2015). MICAz Data Sheet. Available: http://www.openautomation.net/uploadsproductos/micaz_datasheet.pdf."

[45] D. E. Johnson, J. R. Johnson, J. L. Hilburn, and P. D. Scott, *Electric circuit analysis* vol. 3: Prentice Hall, 1997.

[46] A. S. Tanenbaum, "Computer Networks, 4-th Edition," ed: Prentice Hall, 2003.

[47] R. Jain, *The art of computer systems performance analysis*: John Wiley & Sons, 2008.

[48] H. Chang, S.-Y. Chung, and S. Kim, "Interference Channel With a Causal Relay Under Strong and Very Strong Interference," *IEEE Transactions on Information Theory,* vol. 60, pp. 859-865, 2014.

[49] N. Bisnik and A. A. Abouzeid, "Queuing network models for delay analysis of multihop wireless ad hoc networks," *Ad Hoc Networks,* vol. 7, pp. 79-97, 2009.

Vitae

Name	:	Emad Ahmad Mahmoud Aldalu
Nationality	:	Palestinian
Date of Birth	:	March 22 1987
Email	:	emad.a.aldalu@gmail.com
Address	:	Nablus-Palestine

Academic Background:

- MSc in Computer Networks, KFUPM, 2015
- Bachelor in Computer Systems Engineering, Birzeit University, 2010

Publications:

- Accepted conference paper: **E. Aldalu and U. Baroudi, "Does AODV Perform Better with Energy Harvesting?", in "*IWCMC 2015 Wireless Sensor Symposium*", IEEE 2015**

Professional experience:

- Automation test engineer, Exalt Technologies, Ramallah, from July 2010 to January 2013.